

On Fulton's Algorithm for Computing Intersection Multiplicities

Steffen Marcus¹ and Marc Moreno Maza² and Paul Vrbik²

¹ Department of Mathematics, University of Utah

² Department of Computer Science, University of Western Ontario

Abstract. As pointed out by Fulton in his *Intersection Theory*, the intersection multiplicities of two plane curves $V(f)$ and $V(g)$ satisfy a series of 7 properties which *uniquely* define $I(p; f, g)$ at each point $p \in V(f, g)$. Moreover, the proof of this remarkable fact is constructive, which leads to an algorithm, that we call Fulton's Algorithm. This construction, however, does not generalize to n polynomials f_1, \dots, f_n . Another practical limitation, when targeting a computer implementation, is the fact that the coordinates of the point p must be in the field of the coefficients of f_1, \dots, f_n . In this paper, we adapt Fulton's Algorithm such that it can work at any point of $V(f, g)$, rational or not. In addition, we propose algorithmic criteria for reducing the case of n variables to the bivariate one. Experimental results are also reported.

1 Introduction

Intuitively, the intersection multiplicity of two plane curves counts the number of times these curves intersect. There are more formal ways to define this number. The following one is commonly used, see for instance [9, 11, 12, 6, 18]. Given an arbitrary field k and two bivariate polynomials $f, g \in k[x, y]$, consider the affine algebraic curves $C := V(f)$ and $D := V(g)$ in $\mathbb{A}^2 = \bar{k}^2$, where \bar{k} is the algebraic closure of k . Let p be a point in the intersection. The *intersection multiplicity* of p in $V(f, g)$ is defined to be

$$I(p; f, g) := \dim_{\bar{k}}(\mathcal{O}_{\mathbb{A}^2, p} / \langle f, g \rangle)$$

where $\mathcal{O}_{\mathbb{A}^2, p}$ and $\dim_{\bar{k}}(\mathcal{O}_{\mathbb{A}^2, p} / \langle f, g \rangle)$ are the local ring at p and the dimension of the vector space $\mathcal{O}_{\mathbb{A}^2, p} / \langle f, g \rangle$. The intersection multiplicity of two plane curves at a point admits many properties. Among them are the seven below, which are proved in [9, Section 3-3] as well as in [11, 12].

- (2-1) $I(p; f, g)$ is a non-negative integer for any C, D , and p such that C and D have no common component at p . We set $I(p; f, g) = \infty$ if C and D have a common component at p .
- (2-2) $I(p; f, g) = 0$ if and only if $p \notin C \cap D$.
- (2-3) $I(p; f, g)$ is invariant under affine change of coordinates on \mathbb{A}^2 .
- (2-4) $I(p; f, g) = I(p; g, f)$.

(2-5) $I(p; f, g)$ is greater or equal to the product of the multiplicity (see [9, §3.1]) of p in f and g , with equality occurring if and only if C and D have no tangent lines in common at p .

(2-6) $I(p; f, gh) = I(p; f, g) + I(p; f, h)$ for all $h \in k[x, y]$.

(2-7) $I(p; f, g) = I(p; f, g + hf)$ for all $h \in k[x, y]$.

Remarkably, Properties (2-1) through (2-7) uniquely determine $I(p; f, g)$. This observation is made by Fulton in [9, Section 3-3] where he exhibits an algorithm for computing $I(p; f, g)$ using (2-1) through (2-7) as rewrite rules.

In order to obtain a practical implementation of this algorithm, a main obstacle must be overcome. To understand it, let us first recall that computer algebra systems efficiently manipulate multivariate polynomials whenever their coefficients are in the field of rational numbers or in a prime field. In particular, popular algorithms for decomposing the algebraic variety $V(f_1, \dots, f_n)$ with $f_1, \dots, f_n \in k[x_1, \dots, x_n]$ rely only on operations in the field k , thus avoiding to manipulate non-rational numbers, that is, elements of $\bar{k} \setminus k$. For instance, algorithms such as those of [4] represent the variety $V(f_1, \dots, f_n)$ (which is a subset of \bar{k}^n) with finitely many regular chains T_1, \dots, T_e of $k[x_1, \dots, x_n]$ such that we have

$$V(f_1, \dots, f_n) = V(T_1) \cup \dots \cup V(T_e). \quad (1)$$

Now, observe that the intersection multiplicity $I(p; f_1, \dots, f_n)$ of f_1, \dots, f_n at a point p is truly a local notion, while each of the $V(T_i)$ may consist of more than one point, even if T_i generates a maximal ideal of $k[x_1, \dots, x_n]$. Therefore, in order to use regular chains for computing intersection multiplicities, one needs to be able to compute “simultaneously” all the $I(p; f_1, \dots, f_n)$ for $p \in V(T_i)$.

In Section 5 we propose an algorithm achieving the following task in the bivariate case: given $\mathcal{M} \subset k[x, y]$ a maximal ideal, compute the common value of all $I(p; f, g)$ for $p \in V(\mathcal{M})$. In Section 6, we relax the assumption of \mathcal{M} being maximal and require only that a zero-dimensional regular chain $T \subset k[x, y]$ generates \mathcal{M} . However, in this case, the values of $I(p; f, g)$ for $p \in V(T)$ may not be all the same. This situation is handled via splitting techniques as in [4].

Thus, for $n = 2$, we obtain a procedure `TriangularizeWithMultiplicity`(f_1, \dots, f_n) which returns finitely many pairs $(T_1, m_1), \dots, (T_e, m_e)$ where $T_1, \dots, T_e \subset k[x_1, \dots, x_n]$ are regular chains and m_1, \dots, m_e are non-negative integers satisfying Equation (1) and for each $i = 1, \dots, e$, we have

$$(\forall p \in V(T_i)) I(p; f_1, \dots, f_n) = m_i. \quad (2)$$

We are also interested in generalizing Fulton’s Algorithm to n multivariate polynomials in n variables—our ultimate goal being an algorithm that realizes the above specification for $n \geq 2$.

We denote by \mathbb{A}^n the n -dimensional affine space over \bar{k} . Let $f_1, \dots, f_n \in k[x_1, \dots, x_n]$ be n polynomials generating a zero-dimensional ideal with (necessarily finite) zero set $V(f_1, \dots, f_n) \subset \mathbb{A}^n$. Let p be a point in the intersection $V(f_1) \cap \dots \cap V(f_n)$, that is, $V(f_1, \dots, f_n)$. The *intersection multiplicity* of p in $V(f_1, \dots, f_n)$ is the generalization of the 2-variable case (as in [6, 18])

$$I(p; f_1, \dots, f_n) := \dim_{\bar{k}}(\mathcal{O}_{\mathbb{A}^n, p} / \langle f_1, \dots, f_n \rangle),$$

where $\mathcal{O}_{\mathbb{A}^n, p}$ and $\dim_{\bar{k}}(\mathcal{O}_{\mathbb{A}^n, p}/\langle f_1, \dots, f_n \rangle)$ are (respectively) the local ring at the point p and the dimension of the vector space $\mathcal{O}_{\mathbb{A}^n, p}/\langle f_1, \dots, f_n \rangle$.

Among the key points in the proof of Fulton’s algorithmic construction is that $k[x_1]$ is a principal ideal domain. Fulton uses Property (2-7) in an elimination process similar to that of the Euclidean Algorithm. Since $k[x_1, \dots, x_{n-1}]$ is no longer a PID for $n \geq 3$, there is no natural generalization of (2-1) through (2-7) to the n -variate setting (up to our knowledge) that would lead to an algorithm for computing $I(p; f_1, \dots, f_n)$.

To overcome this obstacle, at least for some practical examples, we propose an algorithmic criterion to reduce the n -variate case to that of $n - 1$ variables. This reduction requires two hypotheses: $V(f_n)$ is non-singular at p , and the tangent cone of $V(f_1, \dots, f_{n-1})$ at p and the tangent hyperplane of $V(f_n)$ at p meet only at the point p . The second hypothesis ensures that each component of the curve $V(f_1, \dots, f_{n-1})$ meets the hypersurface $V(f_n)$ without tangency at p . This *transversality* assumption yields a reduction from n to $n - 1$ variables proved with Theorem 1.

In Section 7, we discuss this reduction in detail. In particular, we propose a technique which, in some cases, replaces f_1, \dots, f_n by polynomials g_1, \dots, g_n generating the same ideal and for which the hypotheses of the reduction hold. Finally, in Section 8 we give details on implementing the algorithms herein and in Section 9 we report on our experimentation for both the bivariate case and the techniques of Section 7.

We conclude this introduction with a brief review of related works. In [5], the Authors report on an algorithm with the same specification as the above `TriangularizeWithMultiplicity`(f_1, \dots, f_n). Their algorithm requires, however, that the number of input polynomials is 2. In [17], the Authors outline an algorithm with similar specifications as ours. However, this algorithm is not complete, even in the bivariate case, in the sense that it may not compute the intersection multiplicities of all regular chains in a triangular decomposition of $V(f_1, \dots, f_n)$.

In addition, our approach is novel thanks to an important feature which makes it more attractive in terms of performance. We first compute a triangular decomposition of $V(f_1, \dots, f_n)$ (by *any* available method) thus without trying to “preserve” any multiplicity information. Then, once $V(f_1, \dots, f_n)$ is decomposed we work “locally” at each regular chain. This enables us to quickly discover points p of intersection multiplicity one by checking whether the Jacobian matrix of f_1, \dots, f_n is invertible at p . We have observed experimentally that this strategy leads to massive speedup.

2 Regular Chains

In this section, we recall the notions of a regular chain. From now on we assume that the variables of the polynomial ring $k[x_1, \dots, x_n]$ are ordered as $x_n > \dots > x_1$. For a non-constant $f \in k[x_1, \dots, x_n]$, the *main variable* of f is the largest variable appearing in f , while the *initial* of f is the leading coefficient of f w.r.t. the main variable of f . Let $T \subset k[x_1, \dots, x_n]$ be a set of n non constant

polynomials. We say that T is *triangular* if the main variables of the elements of T are pairwise different. Let t_i be the polynomial of T with main variable x_i . We say that T is a (zero-dimensional) *regular chain* if, for $i = 2, \dots, n$ the initial of t_i is invertible modulo the ideal $\langle t_1, \dots, t_{i-1} \rangle$. Regular chains are also defined in positive dimension, see [1, 15].

For any maximal ideal \mathcal{M} of $k[x_1, \dots, x_n]$ there exists a regular chain T generating \mathcal{M} , see [14]. Therefore, for any zero-dimensional ideal \mathcal{I} of $k[x_1, \dots, x_n]$ there exist finitely many regular chains $T_1, \dots, T_e \subset k[x_1, \dots, x_n]$ such that we have $V(\mathcal{I}) = V(T_1) \cup \dots \cup V(T_e)$. Various algorithms, among them those published in [20, 10, 14, 19, 4], compute such decompositions. The `Triangularize` command of the `RegularChains` library [16] in `MAPLE` implements the decomposition algorithm of [4]. This library also implements another algorithm of [4] that we will use in this paper and which is specified hereafter. For a regular chain $T \subset k[x_1, \dots, x_n]$ and a polynomial $p \in k[x_1, \dots, x_n]$, the operation `Regularize(p, T)` returns regular chains $T_1, \dots, T_e \subset k[x_1, \dots, x_n]$ such that we have $V(T) = V(T_1) \cup \dots \cup V(T_e)$ and for all $i = 1, \dots, e$ we have either $V(p) \cap V(T_i) = \emptyset$ or $V(T) \subset V(p)$. We will make use of the following result which can easily be derived from [4]: if `Regularize(p, T)` returns T_1, \dots, T_e , then we have

$$(\forall p \in V(T_i)) \text{Regularize}(p, T_i) = T_i. \quad (3)$$

3 Intersection Multiplicity

As above, let $f_1, \dots, f_n \in k[x_1, \dots, x_n]$ be n polynomials in n variables such that the ideal $\langle f_1, \dots, f_n \rangle$ they generate is zero-dimensional. Let $p \in V(f_1, \dots, f_n)$ and denote the maximal ideal at p by \mathcal{M}_p . When needed, denote the coordinates of p by $(\alpha_1, \dots, \alpha_n)$, so that we have $\mathcal{M}_p = \langle x_1 - \alpha_1, \dots, x_n - \alpha_n \rangle$.

Definition 1. *The intersection multiplicity of p in $V(f_1, \dots, f_n)$ is given by the length of $\mathcal{O}_{\mathbb{A}^n, p} / \langle f_1, \dots, f_n \rangle$ as an $\mathcal{O}_{\mathbb{A}^n, p}$ -module.*

Since we consider \mathbb{A}^n as defined over the algebraically closed field \bar{k} , we know (see, for instance, [8]) that the length of this module is equal to its dimension as a \bar{k} vector space, which is precisely the definition of Section 1. Our algorithm depends on the fact that the intersection multiplicity satisfies a generalized collection of properties similar to (2-1) through (2-7) for the bi-variate case. They are the following:

- (n-1) $I(p; f_1, \dots, f_n)$ is a non-negative integer.
- (n-2) $I(p; f_1, \dots, f_n) = 0$ if and only if $p \notin V(f_1, \dots, f_n)$.
- (n-3) $I(p; f_1, \dots, f_n)$ is invariant under affine change of coordinates on \mathbb{A}^n .
- (n-4) $I(p; f_1, \dots, f_n) = I(p; f_{\sigma(1)}, \dots, f_{\sigma(n)})$ for any $\sigma \in \mathfrak{S}_n$.
- (n-5) $I(p; (x_1 - \alpha_1)^{m_1}, \dots, (x_n - \alpha_n)^{m_n}) = m_1 \dots m_n$, for all non-negative integers m_1, \dots, m_n .
- (n-6) If $g, h \in k[x_1, \dots, x_n]$ make f_1, \dots, f_{n-1}, gh a zero-dimensional, then $I(p; f_1, \dots, f_{n-1}, gh) = I(p; f_1, \dots, f_{n-1}, g) + I(p; f_1, \dots, f_{n-1}, h)$ holds.

(n-7) $I(p; f_1, \dots, f_{n-1}, g) = I(p; f_1, \dots, f_{n-1}, g + h)$ for all $h \in \langle f_1, \dots, f_{n-1} \rangle$.

In order to reduce the case of n variables (and n polynomials) to that of $n-1$ variables (see Section 7) we require an additional property when $n > 2$. Of course, the assumptions necessary for this property may not hold for every polynomial system. However, we discuss in Section 7 a technique that can overcome this limitation for some practical examples.

(n-8) Assume the hypersurface $h_n = V(f_n)$ is non-singular at p . Let v_n be its tangent hyperplane at p . Assume furthermore that h_n meets each component of the curve $\mathcal{C} = V(f_1, \dots, f_{n-1})$ transversely, that is, the tangent cone $TC_p(\mathcal{C})$ intersects v_n only at the point p . Let $h \in k[x_1, \dots, x_n]$ be the degree 1 polynomial defining v_n . Then, we have

$$I(p; f_1, \dots, f_n) = I(p; f_1, \dots, f_{n-1}, h_n).$$

Recall that the tangent cone $TC_p(\mathcal{C})$ can be thought of as the set of tangents given by limiting the secants to \mathcal{C} passing through p . If $g_1, \dots, g_s \in k[x_1, \dots, x_n]$ are polynomials generating the radical of the ideal $\langle f_1, \dots, f_{n-1} \rangle$, then $TC_p(\mathcal{C})$ is also given by $TC_p(\mathcal{C}) = \langle \text{in}(g_1), \dots, \text{in}(g_s) \rangle$ where $\text{in}(g_i)$, for $i = 1, \dots, s$, is the initial form of g_i , that is, the homogeneous component of g_i of the lowest degree.

Theorem 1. $I(p; f_1, \dots, f_n)$ satisfies the properties (n-1) through (n-8).

Proof. For the first seven properties, adapting the proofs of [9, 12] is routine, except for (n-6), and we omit them for space consideration. For (n-6) and (n-8), as well as the others, the reader is referred to our technical report with the same title and available in the Computing Research Repository (CoRR).

4 Expansion of a Polynomial Family about at an Algebraic Set

The tools introduced herein help build an algorithm for computing the intersection multiplicity of f_1, \dots, f_n at any point of $V(f_1, \dots, f_n)$, whenever the ideal $\langle f_1, \dots, f_n \rangle$ is zero-dimensional and when, for $n > 2$, certain hypothesis are met.

Let y_1, \dots, y_n be n new variables with ordering $y_n > \dots > y_1$. Let $F^1, \dots, F^n \in k[x_1, \dots, x_n, y_1, \dots, y_n]$ be polynomials in $x_1, \dots, x_n, y_1, \dots, y_n$ with coefficients in k . We order the monomials in y_1, \dots, y_n (resp. x_1, \dots, x_n) with the lexicographical term order induced by $y_n > \dots > y_1$ (resp. $x_n > \dots > x_1$). We denote by S_{F^1}, \dots, S_{F^n} the respective monomial supports (i.e. the set of monomials with non-zero coefficients) of F^1, \dots, F^n , regarded as polynomials in the variables y_1, \dots, y_n and with coefficients in $k[x_1, \dots, x_n]$. Let i be any integer index in $1, \dots, n$. Write

$$F^i = \sum_{\mu \in S_{F^i}} F_\mu^i \mu, \tag{4}$$

where all F_μ^i are polynomials of $k[x_1, \dots, x_n]$. In particular, the F_1^i represent F_μ^i when $\mu = y_1^0 \cdots y_n^0 = 1$. Denote by $F_{<y_n}^i$ the polynomial of $k[x_1, \dots, x_n][y_1, \dots, y_{n-1}]$ defined by

$$F_{<y_n}^i = \sum_{\substack{\mu \in S_F^i \\ \mathbf{deg}(\mu, y_n) = 0}} F_\mu^i.$$

Let \mathcal{I} be a (proper) ideal of $k[x_1, \dots, x_n]$. We denote by $\text{NF}(f, \mathcal{I})$ the normal form of f w.r.t. the reduced lexicographical Gröbner basis of \mathcal{I} for $x_n > \cdots > x_1$.

Let $p \in \mathbb{A}^n$ with coordinates $\alpha = (\alpha_1, \dots, \alpha_n)$. For a monomial $\mu = y_1^{e_1} \cdots y_n^{e_n}$, we denote by $\text{shift}(\mu, \alpha)$ the polynomial of $\bar{k}[x_1, \dots, x_n]$ defined by

$$\text{shift}(\mu, \alpha) = (x_1 - \alpha_1)^{e_1} \cdots (x_n - \alpha_n)^{e_n}.$$

We denote by \mathcal{M}_α the maximal ideal of $\bar{k}[x_1, \dots, x_n]$ generated by $x_1 - \alpha_1, \dots, x_n - \alpha_n$. When no confusion is possible, we simply write F and f instead of F^i and f_i . We denote by $\text{eval}(F, \alpha)$ the polynomial

$$\text{eval}(F, \alpha) = \sum_{\mu \in S_F} \text{NF}(F_\mu, \mathcal{M}_\alpha) \text{shift}(\mu, \alpha) \quad (5)$$

in $\bar{k}[x_1, \dots, x_n]$. We call this the *specialization of F at α* . Let $W \subset \mathbb{A}^n$ be an algebraic set over k , that is, the zero set $V(P)$ in \mathbb{A}^n of some $P \subset k[x_1, \dots, x_n]$. Finally, consider a family $(f_\alpha, \alpha \in W)$ of polynomials of $\bar{k}[x_1, \dots, x_n]$.

We say that F is an *expansion of f about W* if for every point α of W we have $\mathbf{f} = \mathbf{eval}(F, \alpha)$. More generally, we say that F is an *expansion of the polynomial family $(f_\alpha, \alpha \in W)$ about W* if for every point α of W we have $\mathbf{f}_\alpha = \mathbf{eval}(F, \alpha)$. We conclude this section with a fundamental example of the concepts introduced below. For $\mu = y^{e_1} \cdots y^{e_n}$, we denote by $c(f, \mu)$ the polynomial of $k[x_1, \dots, x_n]$ defined by $c(f, \mu) = \frac{1}{e_1! \cdots e_n!} \frac{\partial^{e_1 + \cdots + e_n} f}{\partial x_1^{e_1} \cdots \partial x_n^{e_n}}$. (One should recognize these as the coefficients in a Taylor expansion.) Let $S_C(f)$ be the set of the $y^{e_1} \cdots y^{e_n}$ monomials such that $e_i \leq \deg(f, x_i)$ holds for all $i = 1, \dots, n$. Then, the polynomial $C(f) = \sum_{\mu \in S_C(f)} c(f, \mu) \mu$ is an expansion of f about W .

5 Computing Intersection Multiplicities of Bivariate Systems: Irreducible Case

We follow the notations introduced in Section 4. Let F^1, \dots, F^n be the expansions of f_1, \dots, f_n about an algebraic set $W \subset \mathbb{A}^n$. In this section, we assume $W = V(\mathcal{M})$ holds for a maximal ideal \mathcal{M} of $k[x_1, \dots, x_n]$ and that $n = 2$ holds.

Theorem 2. *The intersection multiplicity of f_1, f_2 is the same at any point of $V(\mathcal{M})$; we denote it by $I(\mathcal{M}; f_1, f_2)$. Moreover, Algorithm 1 computes this multiplicity from F^1, F^2 by performing arithmetic operations in $k[x_1, x_2]$ only.*

Algorithm 1: $\text{IM}_2(\mathcal{M}; F^1, F^2)$

Input: $F^1, F^2 \in k[x_1, x_2, y_1, y_2]$ and $\mathcal{M} \subset k[x_1, x_2]$ maximal such that F^1, F^2 are expansions of $f_1, f_2 \in k[x_1, x_2]$ about $V(\mathcal{M})$ and $\langle f_1, f_2 \rangle$ is a zero-dimensional ideal.

Output: $I(\mathcal{M}; f_1, f_2)$.

```
1 if  $\text{NF}(F^1, \mathcal{M}) \neq 0$  then
2   return 0;
3 if  $\text{NF}(F^2, \mathcal{M}) \neq 0$  then
4   return 0;
5  $r := \deg(F^1_{<y_2} \bmod \mathcal{M}, y_1)$ ;
6  $s := \deg(F^2_{<y_2} \bmod \mathcal{M}, y_1)$ ;
7 if  $r = 0$  then
8   return  $\text{tdeg}(F^2_{<y_2} \bmod \mathcal{M}, y_1) + \text{IM}_2(\mathcal{M}; \frac{F^1 - F^1_{<y_2}}{y_2}, F^2)$ ;
9 if  $s = 0$  then
10  return  $\text{tdeg}(F^1_{<y_2} \bmod \mathcal{M}, y_1) + \text{IM}_2(\mathcal{M}; F^1, \frac{F^2 - F^2_{<y_2}}{y_2})$ ;
11  $a_1 := \text{lc}(F^1_{<y_2} \bmod \mathcal{M}, y_1)$ ;
12  $a_2 := \text{lc}(F^2_{<y_2} \bmod \mathcal{M}, y_1)$ ;
13 if  $r \leq s$  then
14   let  $b_1 \in k[x_1, x_2]$  such that  $a_1 b_1 \equiv 1 \pmod{\mathcal{M}}$ ;
15    $H := F^2 - a_2 b_1 y_1^{s-r} F^1$ ;
16   return  $\text{IM}_2(\mathcal{M}; F^1, H)$ ;
17 let  $b_2 \in k[x_1, x_2]$  such that  $a_2 b_2 \equiv 1 \pmod{\mathcal{M}}$ ;
18  $H := F^1 - a_1 b_2 y_1^{r-s} F^2$ ;
19 return  $\text{IM}_2(\mathcal{M}; H, F^2)$ ;
```

This first claim in Theorem 2 should not surprise the expert reader. The length of the module $\mathcal{O}_{\mathbb{A}^n, p} / \langle f_1, \dots, f_n \rangle$ over a non-algebraically closed field is not necessarily equal to the dimension as a k vector space, though length equals dimension when the field is algebraically closed. The dimension, however, remains the same over both k and \bar{k} .

Proof. We show that $\text{IM}_2(\mathcal{M}; F^1, F^2)$, as returned by Algorithm 1, computes $I(p; f_1, f_2)$ uniformly for all $p \in V(\mathcal{M})$ and performs operations in $k[x_1, x_n]$ only. Algorithm correctness and termination follows from three claims.

Claim 1: If $I(p; f_1, f_2) = 0$ holds for some $p \in V(\mathcal{M})$, then $\text{IM}_2(\mathcal{M}; F^1, F^2)$ correctly returns 0.

Claim 2: If $I(p; f_1, f_2) > 0$ holds for all $p \in V(\mathcal{M})$, and if either $\deg(F^1_{<y_2} \bmod \mathcal{M}, y_1) = 0$ or $\deg(F^2_{<y_2} \bmod \mathcal{M}, y_1) = 0$ holds, then $\text{IM}_2(\mathcal{M}; F^1, F^2)$ correctly invokes $\text{IM}_2(\mathcal{M}; G^1, G^2)$ where each $G^i \in k[x_1, x_2, y_1, y_2]$ is an expansion of a polynomial family about $V(\mathcal{M})$ such that $\min(\deg(G^1, y_2), \deg(G^2, y_2)) < \min(\deg(F^1, y_2), \deg(F^2, y_2))$.

Claim 3: If $I(p; f_1, f_2) > 0$ holds for all point $p \in V(\mathcal{M})$, and if $\deg(F^1_{<y_2} \bmod \mathcal{M}, y_1) > 0$ and $\deg(F^2_{<y_2} \bmod \mathcal{M}, y_1) > 0$ both hold, then the call $\text{IM}_2(\mathcal{M}; F^1, F^2)$

correctly invokes $\text{IM}_2(\mathcal{M}; G^1, G^2)$ where each $G^i \in k[x_1, x_2, y_1, y_2]$ is an expansion of a polynomial family about $V(\mathcal{M})$ such that $\min(\deg(G_{<y_2}^1, y_1), \deg(G_{<y_2}^2, y_1))$ is strictly less than $\min(\deg(F_{<y_2}^1, y_1), \deg(F_{<y_2}^2, y_1))$.

Proof (of Claim 1). Assume that there is $p \in V(\mathcal{M})$ such that $I(p; f_1, f_2) = 0$ holds. From (2-2), this implies that we have $p \notin V(f_1, f_2)$. Since \mathcal{M} is maximal, we deduce that $W \cap V(f) = \emptyset$ holds. Thus, the intersection multiplicity of f_1, f_2 is null at any point of $V(\mathcal{M})$. Moreover, deciding whether this latter fact holds amounts to testing whether one of $\text{NF}(F_1^1, \mathcal{M}), \text{NF}(F_1^2, \mathcal{M})$ is zero or not, which can be computed in $k[x_1, x_2]$ with a regular chain generating \mathcal{M} .

Remark 1. From now on, we assume that $I(p; f_1, f_2) > 0$ holds for all $p \in V(\mathcal{M})$. Since \mathcal{M} is maximal, this implies that $W \subseteq V(F_1^1)$ and $W \subseteq V(F_1^2)$ both hold. Besides, the ideal \mathcal{M} is one of the associated primes of $\langle f_1, f_2 \rangle \subset k[x_1, x_2]$.

Proof (of Claim 2). Assume that either

$$\deg(F_{<y_2}^1 \bmod \mathcal{M}, y_1) = 0 \quad \text{or} \quad \deg(F_{<y_2}^2 \bmod \mathcal{M}, y_1) = 0$$

holds. Since the role of f_1 and f_2 can be exchanged, using (2-4), we assume that $\deg(F_{<y_2}^1 \bmod \mathcal{M}, y_1) = 0$ holds. Consider any point $\alpha = (\alpha_1, \alpha_2)$ of $V(\mathcal{M})$. Since F_1^1 is null modulo \mathcal{M} , the relation $\deg(F_{<y_2}^1 \bmod \mathcal{M}, y_1) = 0$ implies that the whole polynomial $F_{<y_2}^1$ is actually null modulo \mathcal{M} . Thus, the specialization $\text{eval}(F^1, \alpha)$ can be divided by $x_2 - \alpha_2$. Applying (2-6), we have

$$I(p; f_1, f_2) = I(p; x_2 - \alpha_2, f_2) + I(p; \frac{f_1}{x_2 - \alpha_2}, f_2), \quad (6)$$

where $I(p; x_2 - \alpha_2, f_2)$ is the trailing degree of f_2 evaluated at $x_2 = \alpha_2$ (via (2-5)). Since F^1, F^2 are expansions of f_1, f_2 about $V(\mathcal{M})$, Equation (6) yields

$$\text{IM}_2(\mathcal{M}; F^1, F^2) = \text{tdeg}(F_{<y_2}^2 \bmod \mathcal{M}, y_1) + \text{IM}_2(\mathcal{M}; \frac{F^1 - F_{<y_2}^1}{y_2}, F^2) \quad (7)$$

where $\text{tdeg}(F_{<y_2}^1 \bmod \mathcal{M}, y_1)$ is the trailing degree of $F_{<y_2}^1$ regarded as a polynomial in y_1 with coefficients in the field $k[x_1, x_2]/\mathcal{M}$.

Proof (of Claim 3). We assume that

$$\deg(F_{<y_2}^1 \bmod \mathcal{M}, y_1) > 0 \quad \text{and} \quad \deg(F_{<y_2}^2 \bmod \mathcal{M}, y_1) > 0$$

both hold. Since the role of f_1 and f_2 can be exchanged, using (2-4),

$$\deg(F_{<y_2}^1 \bmod \mathcal{M}, y_1) \leq \deg(F_{<y_2}^2 \bmod \mathcal{M}, y_1)$$

is assumed to hold. Let $a_1, a_2 \in k[x_1, x_2]$ be polynomials and $r \leq s$ be positive integers such that $a_1 y_1^r$ and $a_2 y_1^s$ are the leading terms of $F_{<y_2}^1$ and $F_{<y_2}^2$ regarded as polynomials in y_1 with coefficients in $k[x_1, x_2]/\mathcal{M}$. Since $W \cap V(a_1) = \emptyset$ holds there exists a polynomial $b_1 \in k[x_1, x_2]$ such that we have $a_1 b_1 \equiv 1 \bmod \mathcal{M}$. Define $H := F^2 - a_2 b_1 y_1^{s-r} F^1$. Clearly, this an expansion of a polynomial family $(h_\alpha, \alpha \in V(\mathcal{M}))$ about $V(\mathcal{M})$ such that we have $\text{eval}(H, \alpha) = h_\alpha$ where

$$h_\alpha := f_2 - a_2(\alpha) b_1(\alpha) (x_1 - \alpha_1)^{s-r} f_1. \quad (8)$$

Using (2-7), we have $I(p; f_1, f_2) = I(p; f_1, h_\alpha)$, for all $p \in V(\mathcal{M})$, yielding

$$\text{IM}_2(\mathcal{M}; F^1, F^2) = \text{IM}_2(\mathcal{M}; F^1, H). \quad (9)$$

6 Computing Intersection Multiplicities of Bivariate Systems: Zero-Dimensional Case

The generalization from *irreducible* zero-dimensional algebraic sets $V(\mathcal{M})$ to *arbitrary* ones relies on standard techniques for computing triangular decomposition of polynomial systems (see for instance [20, 10, 14, 19, 4]).

Algorithm 2 is the adaptation of Algorithm 1 for $n = 2$ variables. In this algorithm we use two yet unmentioned methods: LT and Tdeg, and one yet unmentioned language construct: **output**. Similar to Regularize, the call $\text{LT}(F^i, T)$, or leading term of F^i modulo $\langle T \rangle$, returns a list of pairs, (C, a_{F^i}) , where $C \subset k[x_1, x_2]$ is a regular chain and a_{F^i} is the lexicographical leading term of F^i when viewed as a polynomial in $y_1 < y_2$ with coefficients in $k[x_1, x_2]/\langle C \rangle$; moreover the union of $V(C)$'s form a partition of $V(T)$. The specification for TDeg "trailing degree" is analogue. Finally, as we are returning a *sequence* we use the language construct **output** (x, y) to indicate that (x, y) has been added to the sequence that will ultimately be returned.

Theorem 3. *Algorithm 2 terminates and works correctly.*

Proof. We distinguish two cases: Algorithm 2 *does not split* the computations and *does split* the computations. In this proof, $C_1, \dots, C_e \subset$ designate regular chains of $k[x_1, \dots, x_n]$ such that $V(T)$ is the disjoint union of $V(C_1), \dots, V(C_e)$.

Non-splitting case: Assume that $\text{IM}_2(T; F^1, F^2)$ computed by Algorithm 2 does not split the computation, thus returning a single pair (T, m) . Using Relation (3), one can check that $\text{IM}_2(C_i; F^1, F^2)$ returns (C_i, m) , for each $i = 1, \dots, e$. Assume that C_1, \dots, C_e generate maximal ideals. One can check that, when it does not split, Algorithm 2 performs the same computation as Algorithm 1. By virtue of Theorem 2, Algorithm 1 works correctly with input maximal ideals, thus each call $\text{IM}_2(C_i; F^1, F^2)$ correctly returns (C_i, m) . Consequently, $\text{IM}_2(T; F^1, F^2)$ correctly returns (T, m) also, since is the disjoint union of $V(C_1), \dots, V(C_e)$.

Splitting case: From now on, assume now that the call $\text{IM}_2(T; F^1, F^2)$ computed by Algorithm 2 splits and returns pairs $(C_1, m_1), \dots, (C_e, m_e)$, where we no longer assume that C_1, \dots, C_e generate maximal ideals. From the non-splitting case and Relation (3), we know that each call $\text{IM}_2(C_i; F^1, F^2)$ correctly returns (C_i, m) . We conclude again with the fact that $V(T)$

7 Reduction to the bivariate case

We return to the n -variate case, using the same notations as in Sections 3. We discuss how this n -variate case can be reduced to the bivariate one, for which Algorithm 2 computes the intersection multiplicity of two plane curves (without common components) at any point of their intersection.

We start by considering Property (n-8) of Section 3. Let $p \in V(f_1, \dots, f_n)$. Assume the hypersurface $h_n = V(f_n)$ is non-singular at p . Let v_n be its tangent hyperplane at p . Assume furthermore that the tangent cone $TC_p(C)$ intersects

Algorithm 2: $\text{IM}_2(T; F^1, F^2)$

Input: F^1 and F^2 as given in Algorithm 1

Output: Finitely many pairs (T_i, m_i) where $T_i \subset k[x_1, \dots, x_n]$ are regular chains and $m_i \in \mathbb{Z}^+$ such that Equation (1) holds and for all $p \in V(T^i)$ we have $I(p; f_1, \dots, f_n) = m_i$.

```
1 for  $T \in \text{Regularize}(F_1^1, T)$  do
2   if  $F_1^1 \notin \langle T \rangle$  then
3     output( $T, 0$ );
4   else
5     for  $T \in \text{Regularize}(F_1^2, T)$  do
6       if  $F_1^2 \notin \langle T \rangle$  then
7         output( $T, 0$ );
8       else
9         for  $(T, a_{F^1}) \in \text{LT}(F_{<y_2}^1, T)$  do
10          for  $(T, a_{F^2}) \in \text{LT}(F_{<y_2}^2, T)$  do
11            /* Wlog  $\deg(F_{<y_2}^1) \leq \deg(F_{<y_2}^2)$  */
12            if  $a_{F^1} \in \langle T \rangle$  then
13              for  $(T, d) \in \text{TDeg}(F_{<y_2}^2, T)$  do
14                for  $(T, i) \in \text{IM}_2(T, \frac{F^1 - F^1_{<y_2}}{y_2}, F^2)$  do
15                  output( $T, (d + i)$ );
16              else
17                 $H \leftarrow F^2 - a_{F^2} \cdot \text{Inverse}(a_{F^1}^1, T) \cdot F^1$ ;
18                output( $\text{IM}_2(T, F^1, H)$ );
```

v_n only at the point p . Let $h \in k[x_1, \dots, x_n]$ be the degree 1 polynomial defining v_n . Finally, recall (Theorem 1) that $I(p; f_1, \dots, f_n) = I(p; f_1, \dots, f_{n-1}, h)$ holds.

Up to re-numbering the variables, we can assume that the coefficient of x_n in h is non-zero, thus $h = x_n - h'$, where $h' \in k[x_1, \dots, x_{n-1}]$. Hence, we can rewrite the ideal $\langle f_1, \dots, f_{n-1}, h \rangle$ as $\langle g_1, \dots, g_{n-1}, h \rangle$ where g_i is obtained from f_i by substituting x_n with h' . If instead of a point p , we have a zero-dimensional regular chain $T \subset k[x_1, \dots, x_n]$, we use the techniques developed in Sections 5 and 6 to reduce to the case of a point. Assuming $x_1 < \dots < x_n$, this leads to $I(p; f_1, \dots, f_n) = I(T \cap k[x_1, \dots, x_{n-1}]; g_1, \dots, g_{n-1})$.

In practice, this reduction from n to $n - 1$ variables does not always apply. For instance, this is the case for *Ojika* 2 $\subseteq k[x, y, z]$:

$$x^2 + y + z - 1 = x + y^2 + z - 1 = x + y + z^2 - 1 = 0. \quad (10)$$

However, using the equation $x^2 + y + z - 1 = 0$ to eliminate z from the other two, we obtain two bivariate polynomials $f, g \in k[x, y]$. At any point of $p \in V(h, f, g)$ the tangent cone of the curve $V(f, g)$ is independent of z ; in some sense it is

“vertical”. Moreover, at any point of $p \in V(h, f, g)$ the tangent space of $V(h)$ is *not* vertical. Thus, the reduction applies without computing *any* tangent cones.

We conclude this section by explaining how the tangent cone $TC_p(\mathcal{C})$ is computed when the above trick does not apply. For simplicity, assume $\bar{k} = \mathbb{C}$ and assume that none of the $V(f_i)$ are singular at p . For each component \mathcal{G} through p of $\mathcal{C} = V(f_1, \dots, f_{n-1})$, we proceed as follows: There exists a neighborhood B of p such that $V(f_i)$ is not singular at all $q \in (B \cap \mathcal{G}) \setminus \{p\}$, for $i = 1, \dots, n-1$. Let $v_i(q)$ be the tangent hyperplane of $V(f_i)$ at q . Regard $v_1(q) \cap \dots \cap v_{n-1}(q)$ as a parametric variety with the coordinates of q as parameters. Then, we have $TC_p(\mathcal{G}) = v_1(q) \cap \dots \cap v_{n-1}(q)$ when q approaches p , which we compute by a variable elimination process. Finally, $TC_p(\mathcal{C})$ is the union of all the $TC_p(\mathcal{G})$. This approach avoids standard basis computation and extends easily for working with the zero set $V(T)$ of a zero-dimensional regular chain T instead of a point p .

8 Implementation

We have done an implementation in MAPLE that depends heavily on the RegularChains library. As this implementation is sufficiently different from the theoretical algorithm it is meaningful to discuss how we realized it.

These differences can be traced back to a common origin: the data structure simulating the expansions F^i defined in Section 4 for the purpose of the algorithms of Sections 5 and 6. Recall that the expansions F^1, \dots, F^n belong to $k[x_1, \dots, x_n, y_1, \dots, y_n]$ where x_1, \dots, x_n are the variables of the input polynomials f_1, \dots, f_n and where y_1, \dots, y_n are essentially “placeholders”. But our algorithms fundamentally treat F^1, \dots, F^n as vectors, performing only additions and subtractions on them.

While these expansions F^1, \dots, F^n are a nice trick to manipulate “simultaneously” Taylor expansions at several points of a variety, a naïve implementation could suffer from performance bottleneck (hardly surprisingly when doubling the number of variables). In particular, we observe that during the execution of the algorithms, all the partial derivatives of f_1, \dots, f_n may not be needed. Therefore, one may wish to take advantage of *lazy* or *delayed* evaluation.

A structure utilizing delayed computation is well suited for this. To demonstrate why, suppose that \mathcal{F}^i is a data structure implementing F^i such that $\mathcal{F}^i(a_1, \dots, a_n) = F_\mu^i$ for $\mu = y_1^{a_1} \dots y_n^{a_n}$. To determine $\mathcal{F}^i(a_1, \dots, a_n + 1)$ one must only compute $\frac{1}{a_n+1} \frac{\partial \mathcal{F}^i(a_1, \dots, a_n)}{\partial x_n}$. Combining this rule with $\mathcal{F}^i(a_1, \dots, a_{n-1}, 0) = \mathcal{F}^i(a_1, \dots, a_{n-1})$ and $\mathcal{F}^i(0) = f_i$ gives a recursive function whose output matches our specification. We call these “lazy Taylor expansions” (LTEs).

Moreover these LTEs have a very useful property: $\mathcal{F}^i(a_1, \dots, a_{n-1}) \equiv F_{<y_n}^i$. They are also surprisingly straightforward to implement in MAPLE.

Notice that the “data structure” for the LTEs are in fact procedures. Therefore any method processing LTEs, like `Subtract` for instance, will take as input procedures and return a procedure. This notion may be unusual but requires very little overhead (practically undetectable in our experiments). We outline the remaining important methods for our algorithms:

Division by y_n :

$$\frac{\mathcal{F}^i(a_1, \dots, a_n)}{y_n} = \mathcal{F}^i(a_1, \dots, a_n + 1)$$

Multiplication by μ : Let $\mathcal{F}^i(a_1, \dots, a_n) = 0$ if there is i for which $a_i < 0$, then

$$\mathcal{F}^i(a_1, \dots, a_n) \cdot (y_1^{b_1} \cdots y_n^{b_n}) = \mathcal{F}^i(a_1 - b_1, \dots, a_n - b_n)$$

Substitute $\mathbf{y}_n = \mathbf{h}_1 \mathbf{y}_1 + \cdots + \mathbf{h}_{n-1} \mathbf{y}_{n-1}$. For every b_1, \dots, b_n with $b_n > 0$, $\mathcal{F}(b_1, \dots, b_n) \leftarrow 0$ and

$$\begin{aligned} \mathcal{F}(a_1 + k_1, \dots, a_{n-1} + k_{n-1}) &\leftarrow \mathcal{F}(a_1, \dots, a_{n-1}) + \\ &\sum_{k_1 + \cdots + k_{n-1} = b_n} \binom{b_n}{k_1, \dots, k_{n-1}} h_1^{k_1} \cdots h_{n-1}^{k_{n-1}}. \end{aligned}$$

Using these LTEs along with careful, and repeated, invocations of the `RegularChains[Regularize]` command, our algorithms can be realized.

9 Experiments

We have fully implemented the bivariate case, that is, Algorithm 2, on top of the `RegularChains` library in `MAPLE`. As this is the base case for the n -variate algorithm it is of paramount importance that it runs fast and correctly. The n -variate implementation is a work in progress and there is large room for improvements.

We choose to study systems taken from [2] and [13]—a suite of examples used for benchmarking and testing bivariate system solvers. All timings are given in seconds and the base field has characteristic 962592769 in all cases. It should be noted that, despite 962592769 being a so-called FFT-prime, we are *not* using the `FastArithmeticTools` package of the `RegularChains` library. This is because our current implementation is only generic and works in any characteristic. However, some of the systems in [13] are too challenging for being directly solved in characteristic zero without using an approach based on modular, or other advanced, techniques. Results are in Table 1.

We are happy with the results of these experiments for two reasons. First, we could not find an instance where `Triangularize` produced regular chains for which our algorithm `IM2` could not correctly and expeditiously determine the intersection multiplicities. Secondly, applying Property (2-5) from Section 1 to our bivariate code admits a speedup factor in the hundreds. Indeed this property enables us to determine if the intersection multiplicity is one simply by checking the invertibility of the Jacobian of f_1, f_2 modulo the current regular chain.

Our n -variate implementation is based on the techniques discussed in Section 7. As with the bivariate case, our experiments are done in characteristic 962592769. We have taken examples from [7] (a paper on intersection multiplicity) and from [3] (a test suite for benchmarking homotopy solvers). Observe that

Table 1. (LEFT) Input Polynomials (after specialization to bivariate). **(RIGHT)** Experimental results for the bivariate case. Dimension is calculated by MAPLE's PolynomialIdeals-NumberOfSolutions command which gives the number of solutions counted *with* multiplicity. Time(Δ ize) is time required by RegularChains-Triangularize to decompose the system into $N=\#\text{rc's}$ many regular chains and Time(rc_im) = Time(rc_im(rc₁)) + \dots + Time(rc_im(rc_N)): the total time for rc_im, our implementation of Algorithm 3, to determine intersection multiplicities of an entire system.

Label	Name	terms	degree	System	Dim	Time(Δ ize)	#rc's	Time(rc_im)
1	hard_one	30	37	$\langle 1, 3 \rangle$	888	9.7	20	19.2
2	L6_circles	4	24	$\langle 1, 4 \rangle$	1456	226.0	8	9.023
3	spiral29_24	63	52	$\langle 1, 5 \rangle$	1595	169.4	8	25.4
4	tryme	38	59	$\langle 3, 5 \rangle$	1413	22.5	27	28.6
5	challenge_12	49	30	$\langle 4, 5 \rangle$	1781	218.4	9	13.9
6	challenge_12_1	64	40	$\langle 5, 1 \rangle$	1759	113.0	10	15.8
7	compact_surf	52	18	$\langle 6, 8 \rangle$	1680	99.7	12	37.6
8	degree_6_surf	467	42	$\langle 6, 9 \rangle$	2560	299.3	10	22.9
9	mignotte_xy	81	64	$\langle 6, 10 \rangle$	1320	131.9	7	8.4
10	SA_4_4_eps	63	33	$\langle 6, 11 \rangle$	1440	59.8	17	27.5
11	spider	292	36	$\langle 7, 8 \rangle$	1152	32.8	12	16.2
				$\langle 7, 9 \rangle$	756	18.5	16	11.2
				$\langle 8, 9 \rangle$	1984	374.5	10	11.3
				$\langle 8, 10 \rangle$	1362	232.5	7	9.3
				$\langle 8, 11 \rangle$	1256	49.6	17	45.7
				$\langle 9, 11 \rangle$	1792	115.1	16	17.2
				$\langle 10, 11 \rangle$	1180	40.9	17	21.3

Table 2. Experimental results for the n -variate case. Dimension is again the dimension of the vector space $k[x_1, \dots, x_n]/\langle f_1, \dots, f_n \rangle$ and Points is the degree of the variety $V(f_1, \dots, f_n)$. Δ ize and rc_im are the same as in Table 1. Cones and COV give (respectively) the time to calculate the tangent cones or to do a change of variables of the system. Finally, Total is the sum of the previous three columns and Success is the number of points (counted *with* multiplicity) for which the bivariate reduction was success full over the dimension of of the vector space $k[x_1, \dots, x_n]/\langle f_1, \dots, f_n \rangle$.

Name	Dim	Points	Δ ize	Cones	COV	rc_im	Total	Success
Nbody5	99	49	1.60	0.00	0.06	1.90	2.00	51/99
mth191	27	18	0.56	5400.00	0.04	0.01	5400.00	23/27
ojika2	8	5	0.20	8.20	0.13	0.47	8.80	8/8
E-Arnold1	45	30	0.89	1100.00	0.01	1800.00	2900.00	45/45
ShiftedCubes	27	25	0.66	0.00	0.00	0.52	0.52	27/27

the reduction techniques of Section 7 apply successfully for 3 examples and partially for 2 examples. We also note that tangent cone computations are currently a bottleneck. A new algorithm for this task is work in progress.

Acknowledgements. This project has benefited from useful conversations with Dr. Roi Docampo and Dr. Noah Giansiracusa, and was funded, in part, by grants from Maplesoft, MITACS and NSERC of Canada.

References

1. P. Aubry, D. Lazard, and M. Moreno Maza. On the theories of triangular sets. *J. Symb. Comp.*, 28(1-2):105–124, 1999.
2. E. Berberich, P. Emeliyanenko, and M. Sagraloff. An elimination method for solving bivariate polynomial systems: Eliminating the usual drawbacks. *CoRR*, abs/1010.1386, 2010.
3. D. Bini and B. Mourrain. Polynomial test suite. <http://www-sop.inria.fr/saga/POL/>. Accessed: April 1, 2012.
4. C. Chen and M. Moreno Maza. Algorithms for computing triangular decompositions of polynomial systems. In *Proc. ISSAC'11*, pages 83–90. ACM, 2011.
5. J.-S. Cheng and X.-S. Gao. Multiplicity preserving triangular set decomposition of two polynomials. *CoRR*, abs/1101.3603, 2011.
6. D. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Graduate Text in Mathematics, 185. Springer-Verlag, New-York, 1998.
7. B. H. Dayton and Z. Zeng. Computing the multiplicity structure in solving polynomial systems. In *Proceedings of ISSAC '05*, pages 116–123. ACM, 2005.
8. W. Fulton. *Introduction to intersection theory in algebraic geometry*, volume 54 of *CBMS Regional Conference Series in Mathematics*. Published for the Conference Board of the Mathematical Sciences, Washington, DC, 1984.
9. W. Fulton. *Algebraic curves*. Advanced Book Classics. Addison-Wesley, 1989.
10. M. Kalkbrener. A generalized euclidean algorithm for computing triangular representations of algebraic varieties. *J. Symb. Comp.*, 15:143–167, 1993.
11. F. Kirwan. *Complex algebraic curves*, volume 23 of *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 1992.
12. Anthony W. Knapp. *Advanced algebra*. Cornerstones. Birkhäuser Boston Inc., Boston, MA, 2007. Along with a companion volume it Basic algebra.
13. O. Labs. A list of challenges for real algebraic plane curve visualization software. In I. Z. Emiris, F. Sottile, and T. Theobald, editors, *Nonlinear Computational Geometry*, pages 137–164. Springer New York, 2010.
14. D. Lazard. Solving zero-dimensional algebraic systems. *J. Symb. Comp.*, 15:117–132, 1992.
15. F. Lemaire, M. Moreno Maza, W. Pan, and Y. Xie. When does (T) equal $\text{Sat}(T)$? In *Proc. ISSAC'2008*, pages 207–214. ACM Press, 2008.
16. F. Lemaire, M. Moreno Maza, and Y. Xie. The `RegularChains` library. In Ilias S. Kotsireas, editor, *Maple Conference 2005*, pages 355–368, 2005.
17. Y. L. Li, B. Xia, and Z. Zhang. Zero decomposition with multiplicity of zero-dimensional polynomial systems. *CoRR*, abs/1011.1634, 2010.
18. I. R. Shafarevich. *Basic algebraic geometry. 1*. Springer-Verlag, Berlin, second edition, 1994.
19. D. M. Wang. *Elimination Methods*. Springer, 2000.
20. W. T. Wu. A zero structure theorem for polynomial equations solving. *MM Research Preprints*, 1:2–12, 1987.