Sparse Resultant Perturbations *

Carlos D' Andrea[1] and Ioannis Z. Emiris[2]

[1] Departamento de Matemática,
F.C.E y N., UBA (1428) Buenos Aires, Argentina
cdandrea@dm.uba.ar

[2] INRIA Sophia-Antipolis, France, and
Dept. of Informatics & Telecommunications, Univ. of Athens, Greece
Ioannis.Emiris@inria.fr, http://www-sop.inria.fr/galaad/emiris

Abstract. We consider linear infinitesimal perturbations on sparse resultants. This yields a family of projection operators, hence a general method for handling algebraic systems in the presence of "excess" components or other degenerate inputs. The complexity is simply exponential in the dimension and polynomial in the sparse resultant degree. Our perturbation generalizes Canny's Generalized Characteristic Polynomial (GCP) for the homogeneous case, while it provides a new and faster algorithm for computing Rojas' toric perturbation. We illustrate our approach through its Maple implementation applied to specific examples. This work generalizes the linear perturbation schemes proposed in computational geometry and is also applied to the problem of rational implicitization with base points.

1 Introduction

Resultants (or eliminants) offer an efficient approach (in terms of asymptotic complexity) for solving polynomial systems. Given a system of polynomial equations, they eliminate several variables simultaneously and reduce system solving over the complex numbers to univariate polynomial factorization or a matrix eigenproblem.

In [7], we extended the applicability of the sparse (or toric) resultant to non-generic instances. The sparse resultant subsumes the classical (or projective) resultant; for general information see [5,8,20]. Here, we overview this approach, and discuss two geometric applications: first, we show how our sparse resultant perturbations generalize the known linear perturbations in *computational geometry* and, second, how they are directly applicable to the problem of *parametric implicitization with base points*.

In computational geometry, predicates may sometimes make certain assumptions about the input. The treatment of cases violating these assumptions is tedious and intricate, thus seldom included in the theoretical discussion, yet it remains a nontrivial matter for implementors. For instance,

* Both authors partially supported by the IST Programme of the EU as a Shared-cost RTD (FET Open) Project IST-2000-26473 (ECG - Effective Computational Geometry for Curves and Surfaces), and by Action A00E02 of the ECOS-SeTCIP French-Argentina bilateral collaboration.

in constructing convex hulls in d dimensions, the *Orientation* predicate supposes that no $d+1$ points lie on the same $(d-1)$-dimensional hyperplane. A general method for eliminating the need of explicitly dealing with some of these special cases relies on linear infinitesimal perturbations [9,10,18].

The problem of computing the implicit, or algebraic, representation of a curve, surface, or hypersurface, given its rational parametric representation, lies at the heart of several algorithms in computer-aided geometric design and geometric modeling. The problem is harder in the presence of base points, i.e. when the parametric polynomials vanish for any value of the variables in the implicit equation. Several algorithms exist for implicitization, but fewer can handle base points in the case of surfaces or hypersurfaces, so the field is still of active research; see, for instance, [2,12,14,16] and the references thereof.

Let $\mathcal{A} = (\mathcal{A}_0, \ldots, \mathcal{A}_n)$, with $\mathcal{A}_i \subset \mathbb{Z}^n$ be finite sets. We assume that their union spans the affine lattice $\mathbb{Z}^n$; this technical assumption can be eventually removed. Consider a system of generic Laurent polynomials (i.e. with integer exponents) in n variables $x = (x_1, \ldots, x_n)$:

$$f_i(x) = \sum_{a \in \mathcal{A}_i} c_{ia}\ x^a, \quad c_{ia} \neq 0, \quad i = 0, 1, \ldots, n, \tag{1}$$

The *Newton polytope* of a polynomial f_i is the convex hull of the exponent vectors in its support $\mathcal{A}_i$; it provides the sparse counterpart to total degree. The *mixed volume* of n Newton polytopes (or the corresponding polynomials) is denoted by $\mathrm{MV}(\cdot)$. The cornerstone of sparse elimination theory is the theorem stating that the number of isolated roots in $(\mathbb{C}^*)^n$ of $f_1, \ldots, f_n$, is bounded by the mixed volume of their supports. Mixed volume generalizes Bézout's classical bound given by the product of total degrees.

The *sparse resultant* $\mathrm{Res}_{\mathcal{A}}(f_0, \ldots, f_n)$ is a polynomial in $\mathbb{Z}[c_{ia}]$, homogeneous in the coefficients of each f_i, with degree equal to $\mathrm{MV}(f_0, \ldots, f_{i-1}, f_{i+1}, \ldots, f_n)$. The sparse resultant vanishes after a specialization of the coefficients iff the specialized system (1) has a solution in the *toric variety* containing $(\mathbb{C}^*)^n$ and associated to the given Newton polytopes, where $\mathbb{C}^* := \mathbb{C} \setminus \{0\}$. Typically, it may be computed as an irreducible factor of a matrix determinant. For the sake of simplicity, we consider coefficients in a field whose algebraic closure is $\mathbb{C}$, but this field can be arbitrary.

Let us consider system (1) over n variables and m parameters; the latter may be the coefficients or any smaller parameter set on which the coefficients depend. The sparse resultant then defines a *projection operator* from a space of dimension $n+m$ to one of dimension m, and for generic coefficients it does not vanish. It is typically computed as the determinant, or a nontrivial divisor of the determinant of a *resultant matrix.* Such matrices include Sylvester's and Macaulay's formulae; our construction generalizes these matrices. However, there are three cases where this construction fails:

- The resultant may be identically zero at the specialized coefficients due to lack of genericity, usually because the coefficients depend on few parameters.
- The sparse resultant may also vanish because the well-constrained system has infinite solutions in the toric variety, forming "excess components".
- The sparse resultant may be nonzero, but the resultant matrix determinant may be identically zero due to a vanishing extraneous factor.

The dense homogeneous case was settled in [3] by means of a homogeneous GCP. In the sparse context, the only existing approach [17] either requires substantial additional computation or has to perturb all given coefficients.

We settle the sparse case in its generality by an efficient generalization of the GCP. The main contribution of this paper is a simple perturbation scheme, based on a lifting of the polytopes. Only a subset of the input coefficients need be perturbed. It produces sparse projection operators defined by a determinant such that, after specialization, these operators are not identically zero but vanish on roots in the proper components of the variety, including all isolated roots. Thus, we avoid all shortcomings mentioned above. Furthermore,

- the input supports $\mathcal{A}_i$ are not incremented (i.e., each perturbation polynomial has support in the respective $\mathcal{A}_i$), and
- the overall complexity remains simply exponential in n and scales as a function of the sparsity parameters.

The main algorithm can be summarized in a single sentence: add linear terms in a perturbation variable, with random coefficients, to all monomials on the diagonal of any sparse resultant matrix.

The next section overviews the most relevant previous work. A very general context of linear perturbations is defined in section 3 to subsume all existing constructions on resultants. In section 4 we detail an efficient algorithm, trade randomization for efficiency, bound the asymptotic complexity, and derive bounds on the number of roots from these operators. In section 5 we examine our approach through its Maple implementation and some applications. Input perturbations have been applied in computational geometry in what concerns linear objects; section 6.1 shows that our scheme generalizes this approach, since the resultant may express geometric predicates among non-linear objects as well. Section 6.2 shows a direct application of our method to the problem of implicitizing rational parametric (hyper)surfaces in the presence of singularities or base points.

2 Related work

The direct precursor of our work [3] settled the dense (classical) case with complexity $d^{O(n)}$, where d bounds the total degree of the polynomials. In terms of the matrix dimension N, the running time is in $O(N^4)$. By means of the GCP, Canny computes a projection operator which is not identically zero, but vanishes on all proper components of the system $f_i = 0$ in the dense case, i.e. when $\mathcal{A}_i = \{x \in \mathbb{Z}_{\geq 0}{}^n : x_1 + \ldots + x_n \leq d_i\}$. Macaulay's formula for the classical resultant allows computation of this projection operator as a quotient of two matrices both having a linear perturbation in the diagonal, just like the characteristic polynomial of square matrices.

We concentrate on the sparse resultant and its matrix, which generalizes the coefficient matrix of a linear system, Sylvester's matrix and Macaulay's matrix. The matrix is constructed by means of a mixed subdivision of the Minkowski (or vector) sum of the Newton polytopes and their associated lifting to $\mathbb{R}^{n+1}$ [4]. This has been the first algorithm for the sparse resultant to achieve complexity polynomial in its degree and simply exponential in n. But this approach relies on certain genericity assumptions, mainly that the resultant does not vanish identically; this hypothesis is removed in this paper. An additional source of improvement comes from D'Andrea's recent formula of the sparse resultant as the quotient of two determinants, the numerator being defined from a sparse resultant matrix and the denominator being one of its submatrices [6]. This generalizes Macaulay's classical formula.

Degeneracy for the sparse resultant has been studied by Rojas [17], who introduced a toric GCP. A randomized algorithm may simply perturb all coefficients. This is a general technique but contradicts the goal of efficiency which is to minimize the use of the perturbation variable. Our scheme perturbs only the subset of the coefficients appearing on the matrix diagonal. In the worst case, the perturbed matrix determinants defined by Rojas and in this paper have the same degree in the perturbation variable, namely equal to the matrix dimension. Alternatively, there is a deterministic method based on *fillings* of the Newton polytopes. One advantage of fillings, illustrated in section 5, is that the maximum degree of the perturbed determinant in the perturbation variable is usually smaller than with our construction. Unfortunately, finding such fillings requires time exponential in the system dimension, namely in $O(n^{2.616}k^{2n+2})$ arithmetic steps over $\mathbb{Q}$, where k is the total number of support elements [17, Main Thm 3].

The algebraic foundations of the present work, along with establishing the validity of the perturbation, have appeared in [7]. One can find there those proofs of theorems that were omitted here. That article also examined the question of optimality of the perturbations (by decreasing the number of perturbed coefficients) and the hardness of this problem. It would be interesting to re-examine the possibility of reducing the number of perturbed coefficients in the case of implicitization. Several problem-specific perturbation methods exist. One which is connected to resultants and has been applied to the implicitization problem was proposed in [16] in order to minimize the number of perturbed terms.

In [7], certain algorithms for computing the projection operators by means of matrix operations are suggested, most notably based on interpolation, randomization, and the quasi-Toeplitz structure of sparse resultant matrices [11]. Recent advances in determinant computation should largely accelerate these algorithms [15].

3 Sparse projection operators

This section discusses a framework for efficient perturbations, and the combinatorial properties of sparse systems used to define such perturbations. The *raison d'être* of linear perturbations is that they minimize the degree in the perturbation variable, thus reducing complexity. Suppose we have a family $p := (p_0(x) \ldots, p_n(x))$ of Laurent polynomials such that $\mathrm{supp}(p_i) \subset \mathcal{A}_i$, and

$$\mathrm{Res}_{\mathcal{A}}(p_0, \ldots, p_n) \neq 0.$$

Note that the $\mathcal{A}_i$ are fixed at input, so the inclusion of $\mathrm{supp}(p_i)$ is usually proper.

Definition 1. The *Toric Generalized Characteristic Polynomial* associated with the sequence $(p_0, \ldots, p_n)$ (p-GCP) is $C_p(\epsilon) := \mathrm{Res}_{\mathcal{A}}\,(f_0 - \epsilon p_0, \ldots, f_n - \epsilon p_n)$. ϵ may be regarded as a positive infinitesimal indeterminate.

Proposition 1. *$C_p(\epsilon)$ is a polynomial in ϵ of degree $r := \deg\,(\mathrm{Res}_{\mathcal{A}}(f_0, \ldots, f_n))$ and not identically zero. Its leading coefficient is $\mathrm{Res}_{\mathcal{A}}(p_0, \ldots, p_n)$.*

Proof. Because of the homogeneities of the sparse resultant, we have $C_p(\epsilon) = \mathrm{Res}_{\mathcal{A}}\left(\epsilon(\frac{1}{\epsilon}f_0 - p_0), \ldots, \epsilon(\frac{1}{\epsilon}f_n - p_n)\right) = \epsilon^r \mathrm{Res}_{\mathcal{A}}\left(\frac{1}{\epsilon}f_0 - p_0, \ldots, \frac{1}{\epsilon}f_n - p_n\right)$. In order to see that the latter resultant is nonzero, remark that it tends to $\mathrm{Res}_{\mathcal{A}}(p_0, \ldots, p_n)$, which is clearly nonzero, for ϵ sufficiently large.

Example 1 (Dense case). When all polynomials are dense with respect to their total degree, then $\mathcal{A}_i = \{x \in \mathbb{Z}_{\geq 0}{}^n : x_1 + \ldots + x_n \leq d_n\}$. We set $p_0 = 1$ and $p_i := x_i^{d_i}$, for $i = 1, \ldots, n$. Then $\mathrm{Res}_{\mathcal{A}}(1, x_1^{d_1}, \ldots, x_n^{d_n}) = 1$, and $C_p(\epsilon)$ is the Generalized Characteristic Polynomial constructed in [3].

The u-resultant of system $f_1, \ldots, f_n$ is the resultant of the overconstrained system where $f_0 := u_0 + u_1 x_1 + \cdots + u_n x_n$ and $\mathcal{A}_0 = \{0, e_1, \ldots, e_n\}$. We can relax the original requirement that $\mathrm{Res}_{\mathcal{A}}(p_0, \ldots, p_n) \neq 0$:

Proposition 2 (u-resultant). *Consider the u-resultant construction with linear f_0. Require that $\mathrm{Res}_{\mathcal{A}}(f_0, p_1, \ldots, p_n) \neq 0$ and let $p_0 := 0$ in definition 1. Then $C_p(\epsilon)$ is a polynomial in ϵ of degree $\deg(\mathrm{Res}_{\mathcal{A}}(f_0, \ldots, f_n)) - \mathrm{MV}(f_1, \ldots, f_n)$ and not identically zero. Its leading coefficient is $\mathrm{Res}_{\mathcal{A}}(f_0, p_1, \ldots, p_n)$.*

Proof. The proof of proposition 1 can be adapted to this case, where the leading coefficient becomes $\mathrm{Res}_{\mathcal{A}}(f_0, p_1, \ldots, p_n)$ when ϵ tends to infinity. This is nonzero for generic choices of u_i, because it vanishes only if the u_i are such that there are solutions to the system $f_0 = p_1 = \cdots = p_n = 0$.

Example 2 (Rojas' toric GCP). Choose $p_1, \ldots, p_n$ with support in the family $(\mathcal{A}_1, \ldots, \mathcal{A}_n)$, set $p_0 := 0$ and $f_0 := \sum_{a \in \mathcal{A}_0} u_a\, x^a$, where $\{u_a\}$ are parameters. Now $C_p(\epsilon) \not\equiv 0$ if $\mathrm{Res}_{\mathcal{A}}(f_0, p_1, \ldots, p_n) \neq 0$ (see also [17, Main Thm 2.4(1)]). When the overconstrained system has finite solutions in the toric compactification given by their supports, then $C_p(\epsilon)$ is the polynomial called Toric Generalized Characteristic Polynomial in [17]. Checking whether the system has a finite number of solutions may be accomplished by the combinatorial method of fillings or by using random polynomials with the same supports.

The following theorem states the main property of polynomials $C_p(\epsilon)$, and allows us to regard them as a generalization of the classical GCP. We start with $n+1$ polynomials $f_i\,(y_1, \ldots, y_m, x_1, \ldots, x_n)$ with support (viewed as polynomials in $\mathbb{C}\left[x_1^{\pm 1}, \ldots, x_n^{\pm 1}\right]$) contained in $\mathcal{A}_i$.

Let Z be the algebraic set $V\,(f_0, \ldots, f_n) \subset \mathbb{C}^m \times \mathcal{T}$, where $\mathcal{T}$ is the toric variety associated to the respective Newton polytopes. Recall that $\mathcal{T}$ lies in $\mathbb{P}^N$ for some $N \in \mathbb{Z}_{>0}$, so w.l.o.g. we consider $Z \subset \mathbb{C}^m \times \mathbb{P}^N$.

Theorem 1. *Let W be a proper component of Z, so that the dimension of W is $m-1$. Let $C_p(y_1, \ldots, y_m)(\epsilon)$ be the p-GCP of definition 1. Arranging $C_p(\epsilon)$ in powers of ϵ, let $C_{p,k}(y_1, \ldots, y_m)$ be its coefficient of lowest degree in ϵ, namely k. If $\pi_y : \mathbb{C}^m \times \mathcal{T} \to \mathbb{C}^m$ denotes projection on the y_i coordinates, then $C_{p,k}\,(\pi_y(s)) = 0$, for all $s \in W$.*

The proof, detailed in [7, Thm 4.1], is based on the resultant matrix properties demonstrated in [4].

The coefficient $C_{p,k}$ of minimum degree in ϵ in the p-GCP $C_p(\epsilon)$ of definition 1 is a suitable *projection operator,* i.e., a nonzero polynomial in the parameters y (typically the input coefficients), which vanishes when there is a solution of the specialized system (1) in some proper component. $C_{p,k}$ is not identically zero even after the specialization, thus fulfilling the first two conditions set in the Introduction. The $C_{p,k}$ is an analogue of the projection operator obtained in [17]. The difference is that our projection operator may be computed via a more flexible algebraic framework, thus allowing faster computation.

We recall now the mixed subdivision-based method for computing the sparse resultant from a matrix determinant. This discussion will allow us to construct a family $(p_0, \ldots, p_n)$ of sparse polynomials, with nonzero resultant.

Let $m_i := \#\,(\mathcal{A}_i)$, and set $m := \sum_{i=0}^{n} m_i$. For any $\omega := (\omega_0, \ldots, \omega_n) \in \mathbb{R}^m$, $\omega_i \in \mathbb{R}^{m_i}$, $init_\omega(\mathrm{Res}_{\mathcal{A}})$, the *initial term* of $\mathrm{Res}_{\mathcal{A}}(f_0, \ldots, f_n)$ is a well-defined polynomial in the coefficients of $f_0, \ldots, f_n$ which is a monomial if ω is generic [20]. Let us pick a vector $\omega := (\omega_0, \ldots, \omega_n)$, which corresponds to functions $\omega_i : \mathcal{A}_i \to \mathbb{Q}$, for $i = 0, \ldots, n$. We require that $\omega_i(a) > 0$ for all $a \in \mathcal{A}_i,\ i = 0, \ldots, n$, and suppose that ω_i is the restriction of an affine function (in most cases we can take it to be linear). This hypothesis can be removed [20].

Consider the *lifted polytopes* $Q_{i,\omega} := \mathrm{conv}\{(a, \omega_i(a)) : a \in \mathcal{A}_i\} \subset \mathbb{R}^{n+1}$, and their Minkowski sum $Q_\omega := Q_{0,\omega} + \ldots + Q_{n,\omega}$. Taking the *upper* envelope of Q_ω, we get a coherent polyhedral (or mixed) *subdivision* Δ_ω of the Minkowski sum $Q := Q_0 + \ldots + Q_n$, by means of the natural projection $\mathbb{R}^{n+1} \to \mathbb{R}^n$ of Q_ω. Moreover, this projection, constrained to the upper envelope of Q_ω, becomes a bijection. Suppose that ω is sufficiently generic ([4,20]), then we have a tight mixed coherent decomposition, i.e. each facet in the upper envelope is of the form $F_\omega = F_{0,\omega} + \ldots + F_{n,\omega}$ where F_i is a face of Q_i and $n = \dim(F_0) + \ldots + \dim(F_n)$. This implies that at least one summand is zero-dimensional, i.e., a Newton polytope vertex. Furthermore, the sum expressing F_ω maximizes the aggregate lifting of any sum of n faces equal to F_ω; so it is sometimes called an optimal sum. The cell F, (the projection of F_ω) is said to be *mixed of type* i or i*-mixed* if $\dim F_i = 0,\ \dim F_j > 0,\ j \neq i$.

We use the matrix construction based on Δ_ω by fixing a small vector $\delta \in \mathbb{Q}^n$, and set $\mathcal{E} := \mathbb{Z}^n \cap (\delta + Q)$. The *row content* $\mathrm{RC}(F) = (i, a)$ for any cell of Δ_ω will be a pair (i, a) which satisfies: if $F_\omega = F_{0,\omega} + \ldots + F_{n,\omega}$ is the unique facet on the upper envelope of Q_ω projecting to F expressed by this optimal sum, let i be the *largest* index such that $\dim(F_i) = 0,\ F_i = \{a\}$. We also define the row content of all points $p \in F : p + \delta \in \mathcal{E}$ to be $\mathrm{RC}(p) = \mathrm{RC}(F)$.

Definition 2. Consider all monomials corresponding to the *matrix diagonal:* $\mathcal{B}_i := \{a \in \mathcal{A}_i : \mathrm{RC}(p) = (i, a),\ p \in \mathcal{E}\},\ i = 0, \ldots, n$. Then define the perturbation polynomials using a parameter $\lambda \in \mathbb{C}^*$ as $p_{\lambda,i} := \sum_{a \in \mathcal{B}_i} \lambda^{\omega_i(a)} x^a,\ i = 0, \ldots, n$.

Proposition 3. [7] *For all but finite* $\lambda \in \mathbb{C}^*$*, we have that* $\mathrm{Res}_\mathcal{A}(p_\lambda) \neq 0$*, where* $p_\lambda := (p_{\lambda,0}, \ldots, p_{\lambda,n})$

To deterministically guarantee the validity of a given λ, it suffices to test the sparse resultant matrix and its determinant for p_λ.

4 Matrix construction

Methods for computing the GCP as a (nonzero) determinant of a sparse resultant matrix are proposed in this section. We also derive precise bounds on the number of roots for specialized coefficients, and estimate the complexity of our algorithms.

Construct the square matrix $\mathcal{M}_{\omega,\delta} := (m_{p,p'})_{p,p' \in \mathcal{E}}$ with $m_{p,p'} = c_{ia'}$ for $\mathrm{RC}(p) = (i, a)$ and $p - a + a' = p'$. The matrix determinant is generically a nonzero multiple of $\mathrm{Res}_\mathcal{A}$. Its degree in the coefficients of f_i equals the number of rows containing this polynomial's coefficients. The particular definition of RC($\cdot$) implies that the determinant degree in the coefficients of f_0 is $\deg_{f_0} \mathrm{Res}_{\mathcal{A}} = \mathrm{MV}(f_1, \ldots, f_n)$.

Theorem 2. [7] *Let $p_{\lambda,i}$ as in definition 2. Then, for every specialization of $f_0, \ldots, f_n$ and almost all $\lambda \in \mathbb{C}^*$, $\det \mathcal{M}_{\omega,\delta}(f_0 - \epsilon p_{\lambda,0}, f_1 - \epsilon p_{\lambda,1}, \ldots, f_n - \epsilon p_{\lambda,n})$ is a nonzero polynomial in ϵ of degree $\#\mathcal{E}$. This determinant equals $C_{p_\lambda}(\epsilon)$ multiplied by a polynomial in ϵ, not depending on the coefficients of f_0 and $p_{\lambda,0}$, where $C_{p_\lambda}(\epsilon)$ is the p_λ-GCP. In the case of the u-resultant of $(f_1, \ldots, f_n)$, f_0 does not have to be perturbed.*

The obvious construction of p_λ consists in picking a random value for $\lambda \in \mathbb{C}^*$. Thus only a few random bits are needed.

Proposition 4 (Derandomization). *It is possible to remove randomization completely by replacing $\epsilon p_{\lambda,i}$ in the perturbation by $p_{\epsilon,i}$.*

By replacing the value λ by the perturbation parameter ϵ, this construction raises, in general, the perturbation degree in the resultant matrix and the subsequent computation.

Corollary 1. *If some very small probability of error is acceptable, then polynomials $p_{\lambda,i}$ do not have to depend on λ and their coefficients may be chosen completely at random. The probability that this choice be invalid equals the probability that the aggregate coefficient vector lies on the hypersurface defined by the vanishing of the determinant of the perturbation matrix.*

Let ϕ be the trailing coefficient of $\det \mathcal{M}_{\omega,\delta}(\epsilon)$. Then ϕ is a multiple of $C_{p,k}$. By [6], there exists a submatrix of the sparse resultant matrix such that the quotient of their determinants equals the sparse resultant. In any case, ϕ is a polynomial in the u_i indeterminates, in the u-resultant construction, and its factorization gives all the isolated roots of the given well-constrained system. Otherwise, if we have "hidden" some variable x_i in the space of coefficients, ϕ is a polynomial in x_i. and the zero set of $\phi(x_i)$ contains the coordinates of x_i at all isolated roots of the input system. Recall that if the variety of a system of n polynomials in n variables has positive dimension, then both the $u-$resultant and the resultant using hidden variables vanish identically. By theorem 1, it follows:

Corollary 2 (Root counting). *The degree of ϕ in the hidden variable or, in the case of the u-resultant, in any u_i indeterminate, bounds the number of isolated roots of the* specialized *system, which in some cases is tighter than the generic bound available by mixed volume or Bézout's theorem.*

If the degree of ϕ in the coefficients of some polynomial f_j is strictly smaller than the respective degree of $\mathrm{Res}_{\mathcal{A}}$, then the variety of f_0, …, f_{j-1}, f_{j+1}, …, f_n has positive dimension.

Theorem 3 (Complexity). [7] *All operations described in the above algorithm have asymptotic complexity dominated by the complexity of the matrix construction and root approximation algorithms, namely polynomial in* $\deg(\mathrm{Res}_{\mathcal{A}})$ *and exponential with a linear exponent in* n.

5 Experiments

The perturbation has been implemented in Maple and can be found at the webpage of the second author. Below we report on specific applications.

Example 3. Consider the system in [20, Exam. 2.1]:

$$f_0 = \alpha_1 + \alpha_2 x_1^2 x_2^2 + \alpha_3 x_1 x_2^3,\, f_1 = \beta_1 + \beta_2 x_1^2 + \beta_3 x_1 x_2^2,\, f_2 = \gamma_1 x_1^3 + \gamma_2 x_1 x_2.$$

Setting $\delta := (0, -1/3)$, and affine lifting functions $\omega_1(a,b) := -a/2 - b/4 + 1$, $\omega_2(a,b) := 3a - 5b + 7$, $\omega_3(a,b) := 0$ yields

$$p_{\lambda,0} = \lambda + x_1 x_2^3,\ p_{\lambda,1} = \lambda^{13} x_1^2,\ p_{\lambda,2} = x_1^3 + x_1 x_2.$$

An explicit computation shows that $\mathrm{Res}_{\mathcal{A}}(p_\lambda) = \pm\lambda^{93}$. The second simplification rule in section 4 allows to take $p_{\lambda,1} = x_1^2$, hence $\mathrm{Res}_{\mathcal{A}}(p_\lambda) = \pm\lambda^2$. It is clear that $\lambda = 1$ is a good choice, and produces a good perturbation. We could also apply proposition 4 to avoid using λ altogether and, instead, perturb by $\epsilon^2 + \epsilon x_1 x_2^3$, ϵx_1^2, $\epsilon x_1^3 + \epsilon x_1 x_2$.

Consider now specialization

$$\alpha_1 = \beta_3 = -2,\ \alpha_2 = \alpha_3 = \gamma_1 = -\gamma_2 = 1,\ \beta_1 = u,\ \beta_2 = 2 - u.$$

In $\det \mathcal{M}(\epsilon)$, since the constant coefficient is zero, we conclude that the resultant of the input system vanishes for all u, hence the variety has positive dimension; indeed, it contains line $(1,1,u) \simeq \mathbb{C}$. The linear coefficient $\phi(u)$ yields the 6 projections of the isolated roots of the system on the u-axis. $\phi(u)$ can be computed by interpolating on $T = 8$ terms, instead of $\dim \mathcal{M} = 21$ terms.

Example 4. Consider the system in [4]:

$$\begin{aligned} f_0 &= c_{11} + c_{12} x_1 x_2 + c_{13} x_1^2 x_2 + c_{14} x_1, \\ f_1 &= c_{21} x_2 + c_{22} x_1^2 x_2^2 + c_{23} x_1^2 x_2 + c_{24} x_1, \\ f_2 &= c_{31} + c_{32} x_2 + c_{33} x_1 x_2 + c_{34} x_1. \end{aligned}$$

Setting $\delta := (-3/8, -1/8),\ L \geq l^2 > l \gg 0$ and using $\omega_0 = (L-l)x_1 + (L - l^2)x_2, \omega_1 = (L + l^2)x_1 + (L+1)x_2, \omega_2 = (L-1)x_1 + (L+l)x_2$, we have

$$\begin{aligned} p_{\lambda,0} &= 1, \\ p_{\lambda,1} &= \lambda^{4L+2l^2+2} x_1^2 x_2^2 + \lambda^{3L+2l^2+1} x_1^2 x_2, \\ p_{\lambda,2} &= \lambda^{L+l} x_2 + \lambda^{2L+l-1} x_1 x_2 + \lambda^{L-1} x_1; \end{aligned}$$

and

$$\pm\mathrm{Res}_{\mathcal{A}}(p_\lambda) = \lambda^{16L+3l^2+2} - 2\,\lambda^{17L+4l^2+l+2} + \lambda^{18L+5l^2+2l}.$$

Observe that $\lambda = 1$ is a zero of this polynomial, so another value must be chosen. Any random value is valid with very high probability.

In the special case

$$f_0 = 1 + x_1\,x_2 + x_1^2\,x_2 + x_1,\;\; f_2 = 1 + x_2 + x_1\,x_2 + x_1,$$

the sparse resultant vanishes identically since the variety $V(f_0, f_1)$ has positive dimension (it is formed by the union of the isolated point $(1, -1)$ and the line $\{-1\} \times \mathbb{C}$). With $L = 10$ and $l = 3$, we have that the coefficient of ϵ^2 in the perturbed determinant is

$$\phi = -(c_{22}c_{23})(c_{24} - c_{21} + c_{22} - c_{23})(c_{24} + c_{21} - c_{22} + c_{23}),$$

and we can recover in the last two factors the value of f_1 at the isolated zero $(1, -1)$ and the point $(-1, -1)$ in the positive-dimensional component.

Example 5. This is the example of [17, Sect. 3.2]. To the system

$$1 + 2x - 2x^2y - 5xy + x^2 + 3x^3y,\;\; 2 + 6x - 6x^2y - 11xy + 4x^2 + 5x^3y,$$

we add $f_0 := u_1x + u_2y + u_0$, which does not have to be perturbed, by theorem 2. We use function `spares` to construct a 16×16 matrix `Mue` in parameters u_0, u_1, u_2, ϵ. The number of rows per polynomial are, respectively, $4, 6, 6$, whereas the mixed volumes of the 2×2 subsystems are all equal to 4.

```
Mue := spares ([f0,f1,f2], [x,y]):
Due := det(Mue):                 # in u0,u1,u2,e
degree (Due,e);                  # 12
ldg := ldegree(Due,e);           # 1
phi := primpart(coeff(Due,e,ldg)): factor(phi);
```

For certain ω and δ, we have used $p_{\lambda,1} := -3x^2 + x^3y$, $p_{\lambda,2} := 2 + 5x^2$, not depending on any λ but instead with random coefficients. The perturbed determinant has maximum and minimum degree in ϵ, respectively, 12 and 1. Then $\phi(u_i)$ gives two factors corresponding to isolated solutions $(1/7, 7/4)$ and $(1, 1)$: $(49\,u_2 + 4\,u_1 + 28\,u_0)\,(u_2 + u_1 + u_0)$. Another two factors give points on the line $\{-1\} \times \mathbb{C}$ of solutions, but the specific points are very sensitive to the choice of ω and δ. One such choice yields: $(-u_0 + u_1)\,(27\,u_2 + 40\,u_1 - 40\,u_0)$. Rojas obtains a 17×17 matrix whose perturbed determinant has degree only 8 and the lowest term is again linear in ϵ, though different from the one above.

Example 6. In the robot motion planning implementation of Canny's *roadmap algorithm* in [13], numerous "degenerate" systems are encountered. Let us examine a 3×3 system, where we hide x_3 to obtain dense polynomials of degrees

3, 2, 1:

$$\begin{aligned}
f_0 &= 54x_1{}^3 - 21.6x_1{}^2x_2 - 69.12x_1x_2{}^2 + 41.472x_2{}^3 + (50.625 + 75.45x_3)\,x_1{}^2 \\
&\quad + (-92.25 + 32.88x_3)\,x_1x_2 + (-74.592x_3 + 41.4)\,x_2{}^2 + \\
&\quad +(131.25 + 19.04x_3{}^2 - 168x_3)x_1 + \left(-405 + 25.728x_3{}^2 + 126.4x_3\right) x_2 + \\
&\quad +(-108.8\,x_3{}^2 + 3.75\,x_3 + 234.375), \\
f_1 &= -37.725\,x_1{}^2 - 16.44\,x_1x_2 + 37.296\,x_2{}^2 + (-38.08x_3 + 84)\,x_1 + \\
&\quad + (-63.2 - 51.456x_3)\,x_2 + (2.304x_3{}^2 + 217.6x_3 - 301.875), \\
f_2 &= 15\,x_1 - 12\,x_2 + 16\,x_3.
\end{aligned}$$

Our Maple function `spares` applies an optimal perturbation to an identically singular 14×14 matrix in x_3. Now $\det \mathcal{M}_{\omega,\delta}(\epsilon)$ is of degree 14 and the trailing coefficient of degree 2, which provides a bound on the number of affine roots, tighter than Bézout's and the mixed volume, both equal to 6. We obtain

$$\phi(x_3) = \left(x_3 - \frac{1434}{625}\right)\left(x_3 - \frac{12815703325}{21336}\right),$$

the first solution corresponding to the unique isolated solution but the second one is superfluous, hence the variety has dimension zero and degree 1.

6 Geometric applications

6.1 Predicates

Linear infinitesimal perturbations of the input have allowed computational geometers to avoid implementing the singular, or degenerate, configurations. The definition of what constitutes degenerate input depends on the problem and the predicates employed. The present work is a direct generalization of those approaches and in particular of the linear perturbation of [9,10,18].

As formalized in [18], the main requirement in designing a valid perturbation scheme is to define a set of input objects in generic position, whose defining coefficients shall be multiplied by ϵ and then added to the objects on which the predicate is evaluated. Let us take the standard example of $d+1$ lines in d-dimensional space. *Transversality* decides whether they all meet at a point, which happens precisely when the determinant of a $(d+1)$-dimensional matrix $\Delta = [c_{ij}]$ equals zero, where the $c_{ij},\ i,j \in \{1,\ldots,n+1\}$ are the coefficients of the line equations. If $\det \Delta \neq 0$, its sign indicates on which side of the $(d+1)$st line lies the intersection of the other d lines. Applying the linear perturbation scheme of [9,10] we have

$$c_{ij}(\epsilon) = c_{ij} + \epsilon\, i^j, \text{ or } c_{ij}(\epsilon) = c_{ij} + \epsilon\, (i^j \bmod q), \tag{2}$$

for some prime q exceeding the number of input objects. Choosing between these two schemes depends on the predicates that must be handled, the second scheme being less generally applicable.

Evaluation of the perturbed predicate amounts to determining the sign of the trailing coefficient in the ϵ-polynomial representing the determinant of $\Delta(\epsilon)$. Then, $\Delta(\epsilon) = \Delta + \epsilon V$, where the (i, j) entry of V is i_i^j or i_i^j mod q. A merit of these schemes is that the proposed generic set facilitates computation of the perturbed predicate, by making use of Vandermonde matrices like V, and, more importantly, by reducing the evaluation of the perturbed determinant to computing a characteristic polynomial: $\det \Delta(\epsilon) = \det(\Delta + \epsilon V) = \det(-V)\ \det(-V^{-1}\Delta - \epsilon I)$, where I is the identity matrix.

Defining a polynomial system with nonzero sparse resultant satisfies precisely the aforementioned requirement for designing a valid scheme. But why is the resultant matrix expressing a geometric predicate? This shall depend on the applications that computational geometry wishes to tackle, in particular as it aspires to handle non-linear objects of increased algebraic complexity. Observe that existing predicates are instances of resultants, including Transversality, Orientation, and InCircle/InSphere. They are all typically expressed by determinant signs, and these are determinants of resultant matrices. For these simple examples, the sparse and classical projective resultants coincide. For instance, the Voronoi diagram of line segments gives rise to predicates expressed by quadratic polynomials (capturing distance). A resultant represents the precise predicate [1], which tests whether a segment intersects the Voronoi circle tangent to two segments and passing through a point.

When it comes to implementing an algorithm, all predicates must deal with identically perturbed input. This implies that we should anticipate all possible sparse resultant matrices for a given set of objects. In the worst case, we may perturb all monomials using random coefficients. The main limitation is that the projection operator only gives a necessary condition for solvability of the overconstrained system because we compute the projection operator multiplied by an extraneous factor. If we are able to guarantee that this factor has constant sign, then this approach can be useful. In certain cases, it should be possible to factor it out, especially by [6].

6.2 Implicitization with base points

The problem of switching from a rational parametric representation to an implicit, or algebraic, representation of a curve, surface, or hypersurface lies at the heart of several algorithms in computer-aided design and geometric modeling. This section describes a direct application of our perturbation method to the implicitization problem.

Given are rational parametric expressions

$$x_i = p_i(t)/q(t) \in K(t) = K(t_1, \dots, t_n), \quad i = 0, \dots, n,$$

in terms of the polynomials $p_0, \dots, p_n, q$ over some field K of characteristic zero. The implicitization problem consists in computing the smallest algebraic surface in terms of $x = (x_0, \ldots, x_n)$ containing the closure of the image of the parametric map $t \mapsto x$. The most common case is for curve and surface implicitization, namely when $n = 1$ and $n = 2$ respectively. Several algorithms exist for this problem, cf. [14,19] and the references thereof.

Implicitization is equivalent to eliminating all parameters t from the polynomials $f_i(t) = p_i(t) - x_i q(t)$, regarded as polynomials in t. The resultant is well-defined for this system, and shall be a polynomial in x, equal to the implicit expression, unless it vanishes. This happens precisely when there exist values of t, known as *base points*, for which the f_i vanish for all x. Different methods have been proposed to handle base points, and the field is still of active research, cf. [2,12,14,16] and the references thereof. These methods include resultant perturbations, since the trailing ϵ-coefficient yields the implicit equation. Our perturbation method applies directly, since the projection operator will contain, as an irreducible factor, the implicit equation. The extraneous factor has to be removed by factorization. Distinguishing the implicit equation from the latter is straightforward by using the parametric expressions to generate points on the implicit surface. The rest of the section illustrates our perturbation in the case of base points.

Example 7. Let us consider the de-homogenized version of a system defined in [2]:

$$p_0 = t_1^2,\ p_1 = t_1^3,\ p_2 = t_2^2,\ q = t_1^3 + t_2^3.$$

It has one base point, namely $(0, 0)$, of multiplicity 4. The toric resultant here does not vanish, so it yields the implicit equation

$$x_2^3 x_1^2 - x_0^3 x_1^2 + 2x_0^3 x_1 - x_0^3.$$

But under the change of variable $t_2 \to t_2 - 1$ the new system has zero toric resultant. The determinant of the perturbed 27×27 resultant matrix has a trailing coefficient which is precisely the implicit equation. The degree of the trailing term is 4, which equals in this case, the number of base points in the toric variety counted with multiplicity.

Example 8. The following problem was suggested by Alicia Dickenstein to the second author. The parametric surface is given by

$$x_j = \prod_{i=1}^{7} \left(b_{i0} + \sum_{k=1}^{2} t_k b_{ik} \right)^{b_{ij}}, \quad j = 0, 1, 2,$$

where B is of dimension 7×3 and specified as follows:

$$B^T = \begin{bmatrix} 1 & 0 & -1 & 0 & 2 & -1 & -1 \\ 0 & 1 & -1 & 2 & 0 & -1 & -1 \\ 1 & 1 & -2 & 1 & 0 & -1 & 0 \end{bmatrix}.$$

Base points forming a component of codimension 1 can be easily removed by canceling common factors in the numerator and denominator of the rational expressions for the x_j's. For base points forming components of codimension 2, one may apply our perturbation scheme. There is a single affine base point $(1, -1)$. Our algorithm constructs a 33×33 matrix, whose determinant has a trailing term of degree 3. The corresponding coefficient has total degree 14 in x_0, x_1, x_2. When factorized, it yields the implicit equation, which is of degree 9.

References

1. C. Burnikel, K. Mehlhorn, and S. Schirra. How to compute the Voronoi diagram of line segments: Theoretical and experimental results. In *Proc. ESA-94, LNCS 855*, pages 227–237, 1994.
2. L. Busé. Residual Resultant over the Projective Plane and the Implicitization Problem. In *Proc. ACM Intern. Symp. on Symbolic & Algebraic Comput.*, pages 48–55, 2001.
3. J. Canny. Generalised characteristic polynomials. *J. Symbolic Computation*, 9:241–250, 1990.
4. J.F. Canny and I.Z. Emiris. A subdivision-based algorithm for the sparse resultant. *J. ACM*, 47(3):417–451, May 2000.
5. D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry.* Number 185 in Graduate Texts in Mathematics. Springer-Verlag, New York, 1998.
6. C. D'Andrea. Macaulay-style formulas for the sparse resultant. *Trans. of the AMS*, 354:2595–2629, 2002.
7. C. D'Andrea and I.Z. Emiris. Computing sparse projection operators. In *Symbolic Computation: Solving Equations in Algebra, Geometry, and Engineering*, volume 286 of *Contemporary Mathematics*, pages 121–139, Providence, Rhode Island, 2001. AMS.
8. I.Z. Emiris. Discrete geometry for algebraic elimination. In *this volume.*
9. I.Z. Emiris and J.F. Canny. A general approach to removing degeneracies. *SIAM J. Computing*, 24(3):650–664, 1995.
10. I.Z. Emiris, J.F. Canny, and R. Seidel. Efficient perturbations for handling geometric degeneracies. *Algorithmica, Special Issue on Computational Geometry in Manufacturing*, 19(1/2):219–242, Sep./Oct. 1997.
11. I.Z. Emiris and V.Y. Pan. Symbolic and numeric methods for exploiting structure in constructing resultant matrices. *J. Symbolic Computation*, 33:393–413, 2002.
12. L. González-Vega. Implicitization of Parametric Curves and Surfaces by Using Multidimensional Newton Formulae. *J. Symbolic Comput.*, 23:137–152, 1997.
13. H. Hirukawa and Y. Papegay. Motion planning of objects in contact by the silhouette algorithm. In *Proc. IEEE Intern. Conf. Robotics & Automation*, San Francisco, April 2000. IEEE Robotics and Automation Society.
14. C.M. Hoffmann, J.R. Sendra, and F. Winkler. *Parametric Algebraic Curves and Applications*, volume 23 of *J. Symbolic Comput.*. Academic Press, 1997.
15. E. Kaltofen and G. Villard. On the Complexity of Computing Determinants. In *Proc. Fifth Asian Symp. Computer Math.*, K. Shirayanagi and K. Yokoyama, eds., Lect. Notes in Computing, vol. 9, pages 13–27, World Scientific, Singapore, 2001.

16. D. Manocha and J. Canny. The implicit representation of rational parametric surfaces. *J. Symbolic Computation*, 13:485–510, 1992.
17. J.M. Rojas. Solving degenerate sparse polynomial systems faster. *J. Symbolic Computation, Special Issue on Elimination*, 28:155–186, 1999.
18. R. Seidel. The nature and meaning of perturbations in geometric computing. In *Proc. 11th Symp. Theoret. Aspects Computer Science, LNCS 775*, pages 3–17. Springer-Verlag, 1994.
19. T.W. Sederberg, R. Goldman, and H. Du. Implicitizing rational curves by the method of moving algebraic curves. *J. Symbolic Comput., Spec. Issue Parametric Algebraic Curves & Appl.*, 23:153–175, 1997.
20. B. Sturmfels. On the Newton polytope of the resultant. *J. of Algebr. Combinatorics*, 3:207–236, 1994.