

# Algebraic algorithms for polynomial system solving and applications

Christos Konaxis\*

National and Kapodistrian University of Athens  
Department of Informatics and Telecommunications  
ckonaxis@di.uoa.gr

**Abstract.** We consider sparse elimination theory in order to describe the Newton polytope of the sparse resultant of a given overconstrained algebraic system, by enumerating equivalence classes of mixed subdivisions. In particular, we consider specializations of this resultant to a polynomial in a constant number of variables, typically up to 3. We sketch an algorithm that avoids computing the entire secondary polytope; our goal is that it examines only the silhouette of this polytope with respect to an orthogonal projection. Moreover, since determinantal formulae are not always possible, the most efficient general method for computing resultants is by rational formulae. We propose a single lifting function which yields a simple method for computing Macaulay-type formulae of sparse resultants, in the case of generalized unmixed systems, where all Newton polytopes are scaled copies of each other. As another application of sparse elimination, we consider rationally parameterized plane curves and determine the vertex representation of the implicit equation's Newton polygon.

## 1 Introduction

In this dissertation we study problems in sparse elimination: rational formulae for sparse resultants via a single lifting function, computation of the Newton polytope of the sparse resultant and of some of its interesting specializations, and sparse implicitization of rational parametric plane curves. The sparse (or toric) resultant captures the structure of the polynomials by combinatorial means and constitutes the cornerstone of sparse elimination theory [5, chap.7],[19,29]. It is an important tool in deriving new, tighter complexity bounds for system solving, Hilbert's Nullstellensatz, and related problems. These bounds depend on the polynomials' Newton polytopes and their mixed volumes, instead of total degree, which is the only parameter in classical elimination theory. In particular, if  $d$  bounds the total degree of each polynomial, the projective resultant has complexity roughly  $d^{O(n)}$ , whereas the sparse resultant is computed in time roughly proportional to the number of integer lattice points in the Minkowski sum of the Newton polytopes.

---

\* Dissertation Advisor: Ioannis Z. Emiris, Professor

The sparse resultant is defined for an overconstrained system of  $n+1$  Laurent polynomials  $f_i \in K[x_1^{\pm 1}, \dots, x_n^{\pm 1}]$ , in  $n$  variables over some coefficient ring  $K$ . It is the unique, up to sign, integer polynomial over  $K$  which vanishes precisely when the system has a root in the toric projective variety  $X$  defined by the supports of  $f_i$ , in which the torus  $(\overline{K})^n$  is a dense subset.

## 2 Preliminaries

We now recall some crucial notions of sparse elimination theory. Given a polynomial  $f$ , its *support*  $\mathcal{A}(f)$  is the set of the exponent vectors corresponding to monomials with nonzero coefficients. Its *Newton polytope*  $\mathcal{N}(f)$  is the convex hull of  $\mathcal{A}(f)$ , denoted  $\text{CH}(\mathcal{A}(f))$ . Newton polytopes are the main tool that allows us to translate algebraic problems into the language of combinatorial geometry. The *Minkowski sum*  $A+B$  of  $A, B \subset \mathbb{R}^n$  is the set  $A+B = \{a+b \mid a \in A, b \in B\} \subset \mathbb{R}^n$ . If  $A, B$  are convex polytopes, then  $A+B$  is also a convex polytope. In what follows we will also denote the support of a polynomial  $f_i$  as  $A_i$  and its Newton polytope as  $Q_i$ .

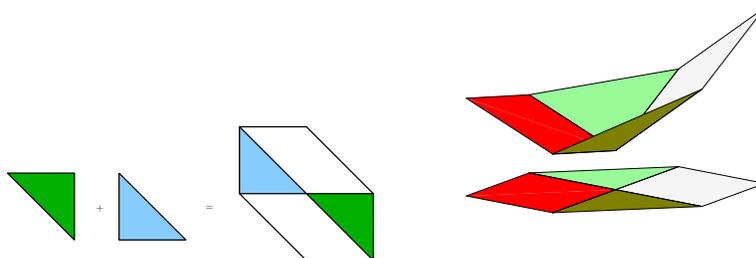
Let  $Q_0, \dots, Q_n$  be polytopes in  $\mathbb{R}^n$  with  $P_i = \text{CH}(A_i)$  and  $Q$  their Minkowski sum. We assume that  $Q$  is  $n$ -dimensional. A *Minkowski cell* of  $Q$  is any full-dimensional convex polytope  $B = \sum_{i=0}^n B_i$ , where each  $B_i$  is a convex polytope with vertices in  $A_i$ . We say that two Minkowski cells  $B = \sum_{i=0}^n B_i$  and  $B' = \sum_{i=0}^n B'_i$  *intersect properly* when the intersection of the polytopes  $B_i$  and  $B'_i$  is a face of both and their Minkowski sum descriptions are compatible.

**Definition 1.** *A mixed subdivision of  $Q$  is any family  $S$  of Minkowski cells which partition  $Q$  and intersect properly as Minkowski sums. A cell  $R$  is mixed, in particular  $i$ -mixed or  $v_i$ -mixed, if it is the Minkowski sum of  $n$  1-dimensional segments  $E_j \subset Q_j$  and one vertex  $v_i \in Q_i$ :  $R = E_0 + \dots + v_i + \dots + E_n$ .*

A mixed subdivision is called *regular* if it is obtained as the projection of the lower hull of the Minkowski sum of lifted polytopes  $\widehat{Q}_i := \{(p_i, \omega_i(p_i)) \mid p_i \in Q_i\}$ . If the lifting function  $\omega := \{\omega_0, \dots, \omega_n\}$  is sufficiently generic, then the induced mixed subdivision is called *fine* or *tight*, and  $\sum_{i=0}^n \dim B_i = \dim \sum_{i=0}^n B_i$ , for every cell  $\sum_{i=0}^n B_i$ . This construction method ensures that the lower hull facets of the Minkowski sum of the lifted polytopes  $\widehat{Q}_i$ , are projected bijectively onto  $Q$ . Thus, every cell  $R$  of the mixed subdivision can be written uniquely as the Minkowski sum  $R = F_0 + \dots + F_n \subset \mathbb{R}^n$ , where each  $F_i$  is a face of  $Q_i$ . Two mixed subdivisions are equivalent if they share the same mixed cells. The equivalence classes are called *mixed cell configurations* [25].

A monomial of the sparse resultant is called *extreme* if its exponent vector corresponds to a vertex of the Newton polytope  $N(\mathcal{R})$  of the resultant. The following corollary of [28, Thm. 2.1], allows us to compute the extreme monomials of the sparse resultant using tight regular mixed subdivisions.

**Corollary 1.** *There exists a surjection from the mixed cell configurations onto the set of extreme monomials of the sparse resultant.*



**Fig. 1.** Minkowski sum of two triangles (left) and construction of its regular mixed subdivision (right).

Given supports  $A_0, \dots, A_n$ , the Cayley embedding  $\kappa$  introduces a new point set  $C := \kappa(A_0, A_1, \dots, A_n) = \bigcup_{i=0}^n (A_i \times \{e_i\}) \subset \mathbb{R}^{2n}$ , where  $e_i$  are an affine basis of  $\mathbb{R}^n$ . The following proposition reduces the computation of regular tight mixed subdivisions to the computation of regular triangulations.

**Proposition 1.** [The Cayley Trick]. *There exists a bijection between the regular tight mixed subdivisions of the Minkowski sum  $P$  and the regular triangulations of  $C$ .*

Regular triangulations are in bijection with the vertices of the *secondary polytope* [19]. A *bistellar flip* is a local modification on a triangulation that leads to a new one. The following theorem allows us to explore the set of regular triangulations of a point set using bistellar flips.

**Theorem 1.** [19] *For every set  $C$  of points affinely spanning  $\mathbb{R}^d$  there is a polytope  $\Sigma(C)$  in  $\mathbb{R}^{|C|-d-1}$ , the secondary polytope of  $C$ , such that its vertices correspond to the regular triangulations of  $C$  and there is an edge between two vertices if and only if the two corresponding triangulations are obtained one from the other by a bistellar flip.*

### 3 Basic results

#### 3.1 Macaulay-type formulae for generalized unmixed sparse resultants

A resultant is most efficiently expressed by a *matrix formula*: this is a generically nonsingular matrix, whose specialized determinant is a multiple of the resultant. Its degree in the coefficients of one polynomial equals the corresponding degree of the resultant. For  $n = 1$  there are matrix formulae named after Sylvester and Bézout, whose determinant equals the resultant. Unfortunately, such determinantal formulae do not generally exist for  $n > 1$ , except for specific cases, e.g. [7,9,22]. Macaulay's seminal result [24] expresses the extraneous factor as a minor

of the matrix formula, for projective resultants of (dense) homogeneous systems, thus yielding the most efficient general method for computing such resultants.

Matrix formulae for the sparse resultant were first constructed in [1]. The construction relies on a lifting of the given polynomial supports, which defines a mixed subdivision of their Minkowski sum into mixed and non-mixed cells, then applies a perturbation  $\delta$  so as to define the integer points that index the matrix. The algorithm was extended in [3,2,28]. In the case of dense systems, the matrix coincides with Macaulay’s numerator matrix.

Extending the Macaulay formula to toric resultants had been conjectured in [3,5,10,19,28]; it was a major open problem in elimination theory. D’Andrea’s result [6] answers the conjecture by a *recursive* definition of a Macaulay-type formula. But this approach does not offer a global lifting, in order to address the stronger original conjecture [10, Conj. 3.1.19], [3, Conj. 13.1].

We give an affirmative answer to this stronger conjecture by presenting a single lifting which constructs Macaulay-type formulae for generalized unmixed systems, i.e. when all Newton polytopes are scaled copies of each other. We state our main result:

**Theorem 2.** [12] *The single lifting algorithm of Section 3.1 constructs a Macaulay-type formula for the toric resultant of an overconstrained generalized unmixed algebraic system, by means of the lifting function of Definition 5.*

Our method can be generalized to certain mixed systems: those with  $n \leq 3$ , as well as systems that possess sufficiently different Newton polytopes. A single lifting algorithm is conceptually simpler and also easier to implement.

D’Andrea’s [6] recursive construction requires one to associate integer points with cells of every dimension from  $n$  to 1. Our method constructs the matrix formula directly, without recursion, by examining only  $n$ -dimensional cells. These are more numerous than the  $n$ -dimensional cells in [6] but our algorithm defines significantly fewer cells totally. The weakness of our method is to consider extra points besides the input supports. Related implementations have been undertaken in Maple, but cover only the original Canny-Emiris method [3], either standalone<sup>1</sup> or as part of library Multires<sup>2</sup>. We expect that our algorithm shall lead to an efficient implementation of Macaulay-type formulae.

Let  $f_0, \dots, f_n$  be polynomials with supports  $A_0, \dots, A_n \subset \mathbb{Z}^n$  and Newton polytopes

$$Q_0, \dots, Q_n \subset \mathbb{R}^n, \quad Q_i = \text{CH}(A_i),$$

where  $\text{CH}(\cdot)$  denotes convex hull. A monomial with exponent  $a = (a_1, \dots, a_n) \in \mathbb{Z}^n$  shall be denoted as  $x^a$ , where  $x := x_1 \cdots x_n$ .

Our lifting shall induce a regular and fine (or tight) mixed subdivision of the Minkowski sum  $\sum_{i=0}^n Q_i$ . Let  $Z$  be the integer lattice generated by  $\sum_{i=0}^n A_i$ . The Minkowski sum  $\sum_{i=0}^n Q_i$  is perturbed by a vector  $\delta \in \mathbb{Q}^n$ , which is sufficiently small with respect to  $Z$ , and in sufficiently generic position with respect to the

<sup>1</sup> [http://www.di.uoa.gr/~emiris/soft\\_alg.html](http://www.di.uoa.gr/~emiris/soft_alg.html)

<sup>2</sup> <http://www-sop.inria.fr/galaad/logiciels/multires.html>

$Q_i$ . The lattice points in  $\mathcal{E} = Z \cap (\sum_{i=0}^n Q_i + \delta)$  are associated to a unique maximal cell of the subdivision, and this allows us to construct a matrix formula  $M$  whose rows and columns are indexed by these points. In particular, polynomial  $x^{p-a_{ij}} f_i$  fills in the row indexed by the lattice point  $p$  in Definition 2.

**Definition 2.** Let  $p \in \mathcal{E}$  lie in a cell  $F_0 + \dots + F_n + \delta$  of the perturbed mixed subdivision, where  $F_i$  is a face of  $Q_i$ . The row content (RC) of  $p$  is  $(i, j)$ , if  $i \in \{0, \dots, n\}$  is the largest integer such that  $F_i$  equals a vertex  $a_{ij} \in A_i$ .

Our method is based on the matrix construction algorithm of [3,10]. For completeness, we recall the basic steps:

1. Pick (affine) liftings  $H_i : \mathbb{Z}^n \rightarrow \mathbb{R} : A_i \rightarrow \mathbb{Q}$ ,  $i = 0, \dots, n$ .
2. Construct a regular fine mixed subdivision of the Minkowski sum  $\sum_{i=0}^n Q_i$  using liftings  $H_i$ .
3. Perturb the Minkowski sum  $\sum_{i=0}^n Q_i$  by a sufficiently small vector  $\delta \in \mathbb{Q}^n$ , so that integer points in  $\sum_{i=0}^n Q_i + \delta$  belong to a unique cell of the subdivision, and assign *row content* to these points by Definition 2.
4. Construct resultant matrix  $M$  with rows and columns indexed by the previous integer points.

The main idea of both our and D'Andrea's methods is that one point, say  $b_{01} \in Q_0$ , is lifted significantly higher. Then, the 0-summand of all maximal cells is either  $b_{01}$  or a face not containing it. In D'Andrea's case, facets not containing  $b_{01}$  correspond to different subsystems where the algorithm recurses (each time on the integer lattice specified by that subsystem). In designing a unique lifting, the issue is that points appearing in two of these subsystems may be lifted differently in different recursions. To overcome this, we introduce several points  $c_{ijs}$ , each lying in a suitable face of  $Q_i$  indexed by  $s$ , very close (with respect to  $Z$ ) to every  $b_{ij}$ , which is lifted very high at recursion  $i$  by D'Andrea's method. This captures the multiple roles  $b_{ij}$  may assume in every recursion step.

**Single lifting Algorithm.** Our algorithm directly generalizes the one given in [3,10], and is based on the 4 steps described above. We modify step (1) and define a new lifting function; moreover, we describe necessary adjustments to the matrix construction and extend step (4) so as to produce the denominator matrix of the Macaulay-type formula. The following three definitions suffice to specify our algorithm.

We shall use  $\mathcal{E}$  to index the rows (and columns) of the numerator matrix  $M$ , whereas the denominator shall be indexed by points lying in non-mixed cells. We focus on generalized unmixed systems, where

$$Q_i = k_i Q \subset \mathbb{R}^n,$$

for some  $n$ -dimensional lattice polytope  $Q$  and  $k_i \in \mathbb{N}^*$ ,  $i = 0, \dots, n$ . Let the vertices of  $Q$  be  $b_0, \dots, b_{|A|}$ , where  $Q = \text{CH}(A)$ . We shall denote the vertices of each  $Q_i = k_i Q$ , for  $i = 0, \dots, n$ , as  $b_{i1}, \dots, b_{i|A|}$ . Obviously,  $b_{ij} := k_i b_j$ .

**Definition 3.** For  $i = 0, \dots, n-2$ , consider any  $(n-i)$ -dimensional face  $F_s^{(i)} \subset Q$ , where integer  $s$  indexes all such faces. Take any vertex  $b_{ij} \in k_i F_s^{(i)}$ , for any valid  $j \in \mathbb{N}$ . Let  $\delta_{ijs} \in \mathbb{Q}^n$  denote a perturbation vector such that:

1.  $b_{ij} + \delta_{ijs}$  lie in the relative interior of  $k_i F_s^{(i)}$ ,
2. It is sufficiently small compared to lattice  $Z$ , and  $\|\delta_{ijs}\| \ll \|\delta\|$ , where  $\|\cdot\|$  is the Euclidean norm and  $\delta$  as above, and
3. It is sufficiently generic to avoid all edges in the mixed subdivision of  $\sum_{i=0}^n Q_i$ .

For an example of Definition 3 see Figure 2, where the (appropriately translated)  $\delta_{ijs}$ 's are depicted by arrows. We shall use the perturbation vectors of Definition 3 to define extra points *not* contained in the input supports. Condition (2) of Definition 3 implies that, in the mixed subdivision induced by the single lifting function  $\beta$  below, the cells created by the introduction of the extra points will not contain integer points after we perturb the mixed subdivision by  $\delta$ . This can be checked at the end of the construction of the mixed subdivision.

**Definition 4.** We define points  $c_{ijs} \in Q_i \cap \mathbb{Q}^n$ , for  $i = 0, \dots, n-2$ . Firstly, set  $c_{011} := b_{01} + \delta_{011} \in Q_0 \cap \mathbb{Q}^n$  where  $\delta_{011}$  satisfies Definition 3. Now let  $\{c_{ijs} \in k_i F_s^{(i)}\}$  be the set of points defined in  $Q_i$ , where  $s$  ranges over all  $(n-i)$ -dimensional faces  $F_s^{(i)} \subset Q$  and  $j$  over the set of indices of points in  $Q_i$ . Then, let  $F_u^{(i+1)}$  be a facet of  $F_s^{(i)}$  such that:

1.  $k_i F_u^{(i+1)}$  does not contain any of the  $b_{ij}$ 's corresponding to the already defined  $c_{ijs}$ 's, and
2.  $k_{i+1} F_u^{(i+1)}$  does not contain any of the already defined  $c_{(i+1)l}$ 's.

For each such facet choose a vertex  $b_{(i+1)j} \in A_{i+1}$ , for some  $j$ , and a suitable perturbation vector  $\delta_{(i+1)ju}$  satisfying Definition 3, and set  $c_{(i+1)ju} := b_{(i+1)j} + \delta_{(i+1)ju} \in Q_{i+1} \cap \mathbb{Q}^n$ .

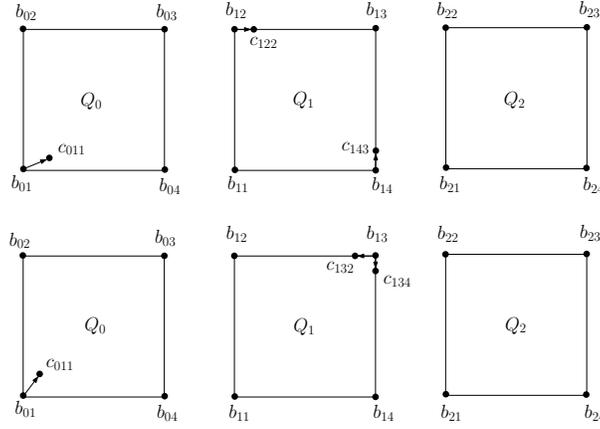
The previous definition implies a many-to-one mapping from the set of  $c_{ijs}$ 's to that of  $b_{ij}$ 's; it reduces to a bijection when restricted to a fixed face  $k_i F_s^{(i)} \subset Q_i$  containing  $b_{ij}$ . For an application of Definition 4 for  $n = 2$  see Figure 2 where  $Q$  is the unit square. In this example, for illustration purposes, we define points  $c_{ijs}$  also on edges of polytope  $Q_1$ .

**Definition 5.** Let  $h_0 \gg h_1 \gg \dots \gg h_{n-1} \gg 1$ . The single lifting algorithm uses sufficiently random linear functions  $H_i, i = 0, \dots, n$ , such that:

$$1 \gg H_i(a_{ij}) > 0, \text{ and } H_i \gg H_t, \quad i < t,$$

where  $a_{ij} \in A_i$  and  $i, t = 0, \dots, n, j = 1, \dots, |A_i|$ . Define a global lifting  $\beta$  as follows:

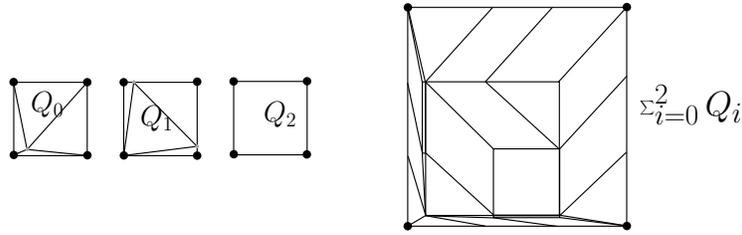
1.  $c_{ijs} \mapsto h_i, c_{ijs} \in k_i F_s^{(i)} \subset Q_i, i = 0, \dots, n-1$ ; this is called *primary lifting*.
2.  $a_{ij} \mapsto H_i(a_{ij}), a_{ij} \in A_i, i = 0, \dots, n$ .



**Fig. 2.** Two scenarios of an application of Def. 4 for 3 unit squares. Facets are numbered clockwise starting from the left vertical edge

Let  $F^\beta$  denote face  $F$  lifted under  $\beta$ . Now  $c_{tjs}^\beta$ , for all valid  $j, s$ , is much higher, respectively lower, than any  $c_{ijs}^\beta$ , for  $i > t$ , respectively  $i < t$ . The  $\beta$ -induced subdivision contains edges with one or two vertices among the  $c_{ijs}$ , and edges from the  $Q_i$ . The vertex set of the upper hull of  $Q_i^\beta$  contains some or all of the  $c_{ijs}^\beta$  and the lifted vertices of  $Q_i$ .

Figure 3 shows the mixed subdivisions of three unit squares and their Minkowski sum, induced by lifting  $\beta$ . Here, the perturbation vectors are not sufficiently small compared to  $\mathbb{Z}^2$  for illustration purposes.



**Fig. 3.** The mixed subdivisions of 3 unit squares and their Minkowski sum induced by lifting  $\beta$

The matrix formula  $M$  constructed by our algorithm is indexed by all lattice points in  $\mathcal{E}$ . To decide the content of each row, every point is associated to a unique (maximal) cell of the mixed subdivision according to Definition 2. The  $t$ -mixed cells contain lattice points as follows:

$$p \in k_0 E_0 + \cdots + k_{t-1} E_{t-1} + c_{tjs} + k_{t+1} E_{t+1} + \cdots + k_n E_n \cap Z,$$

for edges  $E_i \subset Q$  spanning  $\mathbb{R}^n$ . This gives unique writing

$$p = p_0 + \cdots + p_{t-1} + (b_{tj} + \delta_{tjs}) + p_{t+1} + \cdots + p_n, \quad p_i \in A_i \cap E_i.$$

Hence, the row indexed by  $p$ , as with matrix constructions in [3,6], contains a multiple of  $f_t(x)$ :

$$x^{p_0 + \cdots + p_{t-1} + p_{t+1} + \cdots + p_n} f_t(x),$$

and the diagonal element is the coefficient of the monomial with exponent  $b_{tj}$  in  $f_t(x)$ . Similarly, for the rows corresponding to lattice points in non-mixed cells. The extraneous factor  $\det M/\text{Res}(f_0, \dots, f_n)$  is the minor of  $M$  indexed by points in  $\mathcal{E}$  lying in non-mixed cells.

### 3.2 The Newton polygon of rational parametric plane curves

Implicitization is the problem of switching from a parametric representation of a hypersurface to an algebraic one. It is a fundamental question with several applications, see [20]. We consider the implicitization problem for a planar curve, where the polynomials in its parameterization have fixed Newton polytopes. We determine the vertices of the Newton polygon of the implicit equation, or *implicit polygon*, without computing the equation, under the assumption of *generic* coefficients relative to the given supports, i.e. our results hold for all coefficient vectors in some open dense subset of the coefficient space. The support of the implicit equation, or *implicit support*, is taken to be all interior points inside the implicit polygon.

This problem was posed in [32] but has received much attention lately. According to [30], “apriori knowledge of the Newton polytope would greatly facilitate the subsequent computation of recovering the coefficients of the implicit equation [...] This is a problem of numerical linear algebra ...”.

Previous work includes [15,16], where an algorithm constructs the Newton polytope of any implicit equation. That method had to compute all mixed subdivisions, then applies Cor. 1. In [19, chapter 12], the authors study the resultant of two univariate polynomials and describe the facets of its Newton polytope. In [18], the extreme monomials of the Sylvester resultant are described. The approaches in [15,19] cannot exploit the fact that the denominators in a rational parameterization may be identical.

By employing tropical geometry, [30,31] compute the implicit polytope for any hypersurface parameterized by Laurent polynomials. Their theory extends to arbitrary implicit ideals. They give a generically optimal implicit support; for curves, the support is described in [30, example 1.1].

More recently, in [17] the problem was solved in an abstract way by means of composite bodies and mixed fiber polytopes. In [8] the normal fan of the implicit polygon is determined. This is computed by the multiplicities of any parameterization of the rational plane curve. The authors reduce the problem to studying the support function of the implicit polytope and counting the number of solutions of a certain system of equations. The latter is solved by applying a refinement of the Kushnirenko-Bernstein formula for the computation of the

isolated roots of a polynomial system in the torus, given in [26]. As a corollary, they obtain the optimal implicit polygon in the case of generic coefficients.

In [13], we presented a method to compute the vertices of the implicit polygon of polynomial or rational parametric curves, when the denominators differ. We also introduced a method and gave partial results for the case when denominators are equal; both methods are described in final form in [14].

Our main contribution is to determine the vertex structure of the implicit polygon of a rational parameterized planar curve, or implicit vertices, under the assumption of *generic* coefficients. If the coefficients are not sufficiently generic, then the computed polygon contains the implicit polygon. Our approach considers the symbolic resultant which eliminates the parameters and, then, is specialized to yield an equation in the implicit variables. In the case of rationally parameterized curves with different denominators (which includes the case of Laurent polynomial parameterizations), the Cayley trick reduces the problem to computing regular triangulations of point sets in the plane. If the denominators are identical, two-dimensional mixed subdivisions are examined; we show that only subdivisions obtained by *linear* liftings are relevant. These results also apply if the two parametric expressions share the same numerator, or the numerator of one equals the denominator of the other. We prove that, in these cases, only extremal terms matter in determining the implicit polygon as well as in ensuring the genericity hypothesis on the coefficients.

The following proposition collects our main corollaries regarding the shape of the implicit polygon in terms of corner cuts on an initial polygon. A corner cut on a polygon  $P$  is a line that intersects the polygon, excluding one vertex while leaving the rest intact.  $\phi$  is the implicit equation and  $N(\phi)$  is the implicit polygon.

**Proposition 2.**  *$N(\phi)$  is a polygon with one vertex at the origin and two edges lying on the axes. In particular, for polynomial parameterizations,  $N(\phi)$  is a right triangle with at most one corner cut, which excludes the origin. For rational parameterizations with equal denominators,  $N(\phi)$  is a right triangle with at most two cuts, on the same or different corners. For rational parameterizations with different denominators,  $N(\phi)$  is a quadrilateral with at most two cuts, on the same or different corners.*

*Example 1.* Consider the plane curve parameterized by:

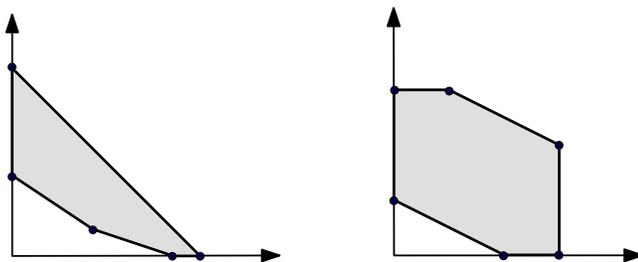
$$x = \frac{t^6 + 2t^2}{t^7 + 1}, y = \frac{t^4 - t^3}{t^7 + 1},$$

Our formulas yield vertices  $(7, 0)$ ,  $(0, 7)$ ,  $(0, 3)$ ,  $(3, 1)$ ,  $(6, 0)$ , which define the actual implicit polygon (see Figure 4, left). Changing the coefficient of  $t^2$  to -1, leads to an implicit polygon with four cuts which is contained in the polygon predicted by our results. This shows the importance of the genericity condition on the coefficients of the parametric polynomials.

An instance where the implicit polygon has 6 vertices is:

$$x = \frac{t^3 + 2t^2 + t}{t^2 + 3t - 2}, y = \frac{t^3 - t^2}{t - 2}.$$

Our results yield implicit vertices  $(0, 1), (0, 3), (3, 0), (1, 3), (2, 0), (3, 2)$  which define the actual implicit polygon (see Figure 4, right).



**Fig. 4.** The implicit polygons of the curves of Example 1

### 3.3 The Newton polytope of the resultant and its specializations

We describe algorithms to compute the Newton polytope of the sparse resultant, or resultant polytope, of an overconstrained system of polynomials. We rely on Corollary 1 and 1 and following [25] it suffices to enumerate a subset of the vertices of the secondary polytope associated with the input data, corresponding to mixed cell configurations. The resultant polytope allows us to compute a superset of the support of the resultant by considering all integer points contained in it; then we can reduce the computation of the resultant to linear algebra [4].

Corollary 1 establishes a surjection from the set of mixed cell configurations onto the set of vertices of the resultant polytope. Experiments<sup>3</sup> indicate that mixed cell configurations are, depending on the input, much less numerous than mixed subdivisions, hence the computation of the resultant vertices becomes more efficient if we focus on the former.

The set of mixed cell configurations corresponds bijectively by the Cayley trick 1, to a set of equivalence classes of regular triangulations. This set can be regarded as a subset of the vertices of the secondary polytope. Thus, we can enumerate mixed cell configurations by enumerating this subset of triangulations. Several algorithms and implementations enumerate regular triangulations e.g. PUNTOS [23], TOPCOM [27], and the algorithm in [21]. We characterize the edges of the secondary polytope that connect the equivalence classes. The sub-graph of the secondary polytope with vertices, regular triangulations corresponding to mixed cell configurations, and the previous edges, is connected.

In [13], we computed the Newton polytope of specialized resultants while avoiding to compute the entire secondary polytope; our approach was to examine the silhouette of the latter with respect to an orthogonal projection. This method is revisited in [11] by studying output-sensitive methods to compute the

<sup>3</sup> See for example the webpage <http://ergawiki.di.uoa.gr/index.php/Implicitization>

resultant polytope. Applications such as the computation of the  $u$ -resultant or implicitization of polynomial parametric curves or surfaces call for the computation of the resultant polytope after a specialization of some of its indeterminates, i.e. some of the coefficients of the input polynomials. This reduces to enumerating the vertices lying on the silhouette of the secondary polytope  $\Sigma(C)$  with respect to some suitably defined projection. For example, the projection of  $\Sigma(C)$  to  $\mathbb{R}^2$  solves the problem of implicitization of polynomial curves, the projection to  $\mathbb{R}^3$  the one of polynomial surfaces etc. The silhouette can be obtained naively by computing all the vertices of  $\Sigma(C)$ , then projecting them to the subspace of smaller dimension. For efficiency we want to enumerate only the vertices lying on a silhouette of  $\Sigma(C)$  with respect to a projection to be defined by the problem, without computing  $\Sigma(C)$ .

In short, we have the following polytope theory problem: We have a high dimensional polytope  $\Sigma(C)$  which we know only locally. By this we mean that from every vertex we have an oracle to find the coordinates of all of its neighbours. We describe an algorithm to compute, for a certain projection  $\pi$  to a space of 1,2, or 3 dimensions, the projection  $\pi(\Sigma(C))$ .

## References

1. J. Canny and I. Emiris. An efficient algorithm for the sparse mixed resultant. In G. Cohen, T. Mora, and O. Moreno, editors, *Proc. Intern. Symp. on Applied Algebra, Algebraic Algor. and Error-Corr. Codes (Puerto Rico)*, number 263 in Lect. Notes in Comp. Science, pages 89–104, Berlin, 1993. Springer-Verlag.
2. J. Canny and P. Pedersen. An algorithm for the Newton resultant. Technical Report 1394, Comp. Science Dept., Cornell University, 1993.
3. J.F. Canny and I.Z. Emiris. A subdivision-based algorithm for the sparse resultant. *J. ACM*, 47(3):417–451, May 2000.
4. R.M. Corless, M.W. Giesbrecht, I.S. Kotsireas, and S.M. Watt. Numerical implicitization of parametric hypersurfaces with linear algebra. In *Artificial intelligence and symbolic computation, (Madrid, 2000)*, pages 174–183. Springer, Berlin, 2001.
5. D. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Number 185 in GTM. Springer, New York, 2nd edition, 2005.
6. C. D’Andrea. Macaulay-style formulas for the sparse resultant. *Trans. of the AMS*, 354:2595–2629, 2002.
7. C. D’Andrea and A. Dickenstein. Explicit formulas for the multivariate resultant. *J. Pure Appl. Algebra*, 164(1-2):59–86, 2001.
8. C. D’Andrea and M. Sombra. The Newton polygon of a rational plane curve. *Mathematics in Computer Science - MCS Special Issue on Computational Geometry and Computer-Aided Design*, 4(1):3–24, 2010.
9. A. Dickenstein and I.Z. Emiris. Multihomogeneous resultant formulae by means of complexes. *J. Symbolic Computation*, 36(3-4):317–342, 2003. Special issue on ISSAC 2002.
10. I.Z. Emiris. *Sparse Elimination and Applications in Kinematics*. PhD thesis, Computer Science Division, Univ. of California at Berkeley, December 1994.
11. I.Z. Emiris, V. Fisikopoulos, and C. Konaxis. Regular triangulations and resultant polytopes. In *Proc. 26th European Workshop on Computational Geometry*, pages 137–140, Dortmund, Germany, 2010.

12. I.Z. Emiris and C. Konaxis. Single-lifting macaulay-type formulae of generalized unmixed sparse resultants. *J. Symbolic Computation*, Elsevier, 2010. Submitted.
13. I.Z. Emiris, C. Konaxis, and L. Palios. Computing the Newton polytope of specialized resultants. In *Proc. Intern. Conf. MEGA (Effective Methods in Algebraic Geometry)*, 2007. [www.ricam.oeaw.ac.at/mega2007/electronic/45.pdf](http://www.ricam.oeaw.ac.at/mega2007/electronic/45.pdf).
14. I.Z. Emiris, C. Konaxis, and L. Palios. Computing the Newton polygon of the implicit equation. *Mathematics in Computer Science - MCS Special Issue on Computational Geometry and Computer-Aided Design*, 4(1):25, 2010.
15. I.Z. Emiris and I.S. Kotsireas. Implicitization with polynomial support optimized for sparseness. In V. Kumar et al., editor, *Proc. Intern. Conf. Comput. Science & Appl. 2003, Montreal, Canada (Intern. Workshop Computer Graphics & Geom. Modeling)*, volume 2669 of *LNCS*, pages 397–406. Springer, 2003.
16. I.Z. Emiris and I.S. Kotsireas. Implicitization exploiting sparseness. In R. Jarnardan, M. Smid, and D. Dutta, editors, *Geometric and Algorithmic Aspects of Computer-Aided Design and Manufacturing*, volume 67 of *DIMACS*, pages 281–298. AMS/DIMACS, 2005.
17. A. Esterov and A. Khovanskii. Elimination theory and Newton polytopes, 2007. [arXiv.org:math/0611107v2](http://arXiv.org/math/0611107v2).
18. I.M. Gelfand, M.M. Kapranov, and A.V. Zelevinsky. Discriminants of polynomials in several variables and triangulations of Newton polytopes. *Leningrad Math. J.*, 2(3):449–505, 1991. (Translated from *Algebra i Analiz* 2, 1990, pp. 1–62).
19. I.M. Gelfand, M.M. Kapranov, and A.V. Zelevinsky. *Discriminants, Resultants and Multidimensional Determinants*. Birkhäuser, Boston, 1994.
20. C.M. Hoffmann. *Geometric and Solid Modeling*. Morgan Kaufmann, 1989.
21. H. Imai, T. Masada, F. Takeuchi, and K. Imai. Enumerating triangulations in general dimensions. *Intern. J. Comput. Geom. Appl.*, 12(6):455–480, 2002.
22. A. Khetan. The resultant of an unmixed bivariate system. *J. Symbolic Computation*, 36:425–442, 2003.
23. J. De Loera. PUNTOS, 1994. [http://www.math.ucdavis.edu/~deloera/RECENT\\_WORK/puntos2000](http://www.math.ucdavis.edu/~deloera/RECENT_WORK/puntos2000).
24. F.S. Macaulay. Some formulae in elimination. *Proc. London Math. Soc.*, 1(33):3–27, 1902.
25. T. Michiels and J. Verschelde. Enumerating regular mixed-cell configurations. *Discr. Comput. Geometry*, 21(4):569–579, 1999.
26. P. Philippon and M. Sombra. A refinement of the Bernstein-Kushnirenko estimate. *Advances in Mathematics*, 218:1370–1418, 2008.
27. Jörg Rambau. TOPCOM: Triangulations of point configurations and oriented matroids. In Arjeh M. Cohen, Xiao-Shan Gao, and Nobuki Takayama, editors, *Mathematical Software—ICMS 2002*, pages 330–340. World Scientific, 2002.
28. B. Sturmfels. On the Newton polytope of the resultant. *J. Algebraic Combin.*, 3:207–236, 1994.
29. B. Sturmfels. *Solving Systems of Polynomial Equations*. Number 97 in CBMS Regional Conference Series in Math. AMS, Providence, RI, 2002.
30. B. Sturmfels, J. Tevelev, and J. Yu. The Newton polytope of the implicit equation. *Moscow Math. J.*, 7(2), 2007.
31. B. Sturmfels and J. Yu. Tropical implicitization and mixed fiber polytopes. In *Software for Algebraic Geometry*, volume 148 of *IMA Volumes in Math. & its Applic.*, pages 111–131. Springer, New York, 2008.
32. B. Sturmfels and J.T. Yu. Minimal polynomials and sparse resultants. In F. Orecchia and L. Chiantini, editors, *Proc. Zero-Dimensional Schemes (Ravello, June 1992)*, pages 317–324. De Gruyter, 1994.