7

Toric resultants and applications to geometric modelling

Ioannis Z. Emiris

Department of Informatics & Telecommunications, National Kapodistrian University of Athens, Panepistimiopolis 15784 Greece, emiris@di.uoa.gr

Summary. Toric (or sparse) elimination theory uses combinatorial and discrete geometry to exploit the structure of a given system of algebraic equations. The basic objects are the Newton polytope of a polynomial, the Minkowski sum of a set of convex polytopes, and a mixed polyhedral subdivision of such a Minkowski sum. Different matrices expressing the toric resultant shall be discussed, and effective methods for their construction will be described based on discrete geometric operations, namely the subdivision-based methods and the incremental algorithm. The former allows us to produce Macaulay-type formulae of the toric resultant by determining a matrix minor that divides the determinant in order to yield the precise resultant. Toric resultant matrices exhibit a quasi-Toeplitz structure, which may reduce complexity by almost one order of magnitude in terms of matrix dimension.

We discuss perturbation methods to avoid the vanishing of the matrix determinant, or of the toric resultant itself, when the coefficients, which are initially viewed as generic, take specialized values. This is applied to the problem of implicitizing parametric (hyper)surfaces in the presence of base points. Another important application from geometric modelling concerns the prediction of the support of the implicit equation, based on toric elimination techniques.

Toric resultant matrices reduce the numeric approximation of all common roots of a polynomial system to a problem in numerical linear algebra. In addition to a survey of recent results, this chapter points to open questions regarding the theory and practice of toric elimination methods.

7.0 Introduction

Toric (or sparse) elimination theory uses combinatorial and discrete geometry to model the structure of a given system of algebraic equations. In particular, we consider algebraic equations with a specific monomial structure. It is thus possible to describe certain algebraic properties of the given system by combinatorial means. This chapter provides a comprehensive state-of-the-art introduction to the theory of toric elimination and toric resultants, paying special attention to the algorithmic and computational issues involved. Different matrices expressing the toric resultants shall be discussed, and effective methods for their construction will be defined based on discrete geometric operations, as well as linear algebra. Toric resultant matrices exhibit a structure close to that of Toeplitz matrices, which may reduce complexity by almost one order of magnitude. These matrices reduce the numeric approximation of all common roots to a problem in numerical linear algebra, as described in Section 7.5 and, in more depth, in Chapters 2 and 3. A relevant feature of resultant matrices in general, is their continuity with respect to small perturbations in the input coefficients.

Our goal is to exploit the fact that systems encountered in engineering applications are, more often than not, characterized by some structure. This claim shall be substantiated by examples in geometric modelling and computer-aided design as well as robotics; further applications exist in vision, and structural molecular biology (cf. [Emi97, EM99b]). A specific motivation comes from systems that must be repeatedly solved for different coefficients, in which case the resultant matrix can be computed exactly once. This occurs, for instance, in parallel robot calibration, see e.g. [DE01c], where 10,000 instances may have to be solved.

This chapter is organized as follows. The next section describes briefly the main steps in the theory of toric elimination, which aspires to generalize the results and algorithms of its mature counterpart, classical elimination. Section 7.2 presents the construction of toric resultant matrices of Sylvester-type. The following section offers a method for implicitizing parametric (hyper)surfaces, including the case of singular inputs, by means of perturbed toric resultants. Section 7.4 applies the tools of toric elimination for predicting the support of the implicit equation. The last section reduces solution of arbitrary algebraic systems to numerical linear algebra, thus yielding methods which avoid any issues of convergence.

This chapter will be of particular interest to graduate students and researchers in theoretical computer science or applied mathematics wishing to combine discrete and algebraic geometry. Some basic knowledge of discrete geometry for polyhedral objects in arbitrary dimension is assumed.

Previous work and open questions are mentioned in the corresponding sections. All algorithms discussed have been implemented either in Maple and/or in `C`, and are publicly available through the author's webpage. Most are also available in the Maple library MULTIRES or the `C++` library SYNAPS, both accessible on the Internet[1].

7.1 Toric elimination theory

Toric elimination generalizes several results of classical elimination theory on multivariate polynomial systems of arbitrary degree by considering their structure. This leads to stronger algebraic and combinatorial results in general

[1] `http://www-sop.inria.fr/galaad/logiciels/`

[CLO98, GKZ94, Stu94a, Stu02]. Assume that the number of variables is n; roots in $(\overline{K}^*)^n$ are called *toric*, where $\overline{K}$ is the algebraic closure of the coefficient field. We use x^e to denote the monomial (or power product) $x_1^{e_1} \cdots x_n^{e_n}$, where $e = (e_1, \ldots, e_n) \in \mathbb{Z}^n$; note that we allow integer exponents. Let the input *Laurent polynomials* be

$$f_1, \ldots, f_n \in K[x_1^{\pm 1}, \ldots, x_n^{\pm 1}]. \tag{7.1}$$

Let the *support* $A_i = \{a_{i1}, \ldots, a_{im_i}\} \subset \mathbb{Z}^n$ denote the set of exponent vectors corresponding to monomials in f_i with nonzero coefficients:

$$f_i = \sum_{j=1}^{m_i} c_{ij} x^{a_{ij}}, \quad \text{for } c_{ij} \neq 0.$$

The *Newton polytope* $Q_i \subset \mathbb{R}^n$ of f_i is the convex hull of support A_i, in other words, the smallest convex polytope that includes all points in A_i. This is a bounded subset of $\mathbb{R}^n$, of dimension up to n. Newton polytopes provide a bridge from algebra to geometry since they permit certain algebraic problems to be cast in geometric terms. For background information and algorithms on polytope theory, the reader may refer to [Ewa96, Sch93]. For arbitrary sets A and $B \subset \mathbb{R}^n$, their *Minkowski sum* is

$$A + B = \{a + b \mid a \in A, b \in B\},$$

where $a + b$ represents the vector sum of points in $\mathbb{R}^n$. For convex polytopes A and B, $A + B$ is a convex polytope.

Definition 7.1.1. *Given convex polytopes* $A_1, \ldots, A_n, A'_k \subset \mathbb{R}^n$, *the* mixed volume $\mathrm{MV}(A_1, \ldots, A_n)$ *is the unique real-valued non-negative function, invariant under permutations, such that,*

$$\mathrm{MV}(A_1, \ldots, \mu A_k + \rho A'_k, \ldots, A_n)$$

is equal to

$$\mu \mathrm{MV}(A_1, \ldots, A_k, \ldots, A_n) + \rho \mathrm{MV}(A_1, \ldots, A'_k, \ldots, A_n),$$

for $\mu, \rho \in \mathbb{R}_{\geq 0}$. *Moreover, we set*

$$\mathrm{MV}(A_1, \ldots, A_n) := n!\, \mathrm{Vol}(A_1), \ \textit{when } A_1 = \cdots = A_n,$$

where $\mathrm{Vol}(\cdot)$ *denotes euclidean volume in* $\mathbb{R}^n$.

If the polytopes have integer vertices, their mixed volume takes integer values. Two equivalent definitions are the following.

Proposition 7.1.2. *For $\lambda_1, \ldots, \lambda_n \in \mathbb{R}_{\geq 0}$ and for convex polytopes $Q_1, \ldots, Q_n$ lying in $\mathbb{R}^n$, the* mixed volume $\mathrm{MV}(Q_1, \ldots, Q_n)$ *is precisely the coefficient of $\lambda_1 \lambda_2 \cdots \lambda_n$ in*

$$\mathrm{Vol}(\lambda_1 Q_1 + \cdots + \lambda_n Q_n),$$

when the latter is expanded as a polynomial in $\lambda_1, \ldots, \lambda_n$. Equivalently,

$$\mathrm{MV}(Q_1, \ldots, Q_n) = \sum_{I \subset \{1,\ldots,n\}} (-1)^{n-|I|} \,\mathrm{Vol}\left(\sum_{i \in I} Q_i\right).$$

In the last equality, I ranges over all subsets of $\{1, \ldots, n\}$, so for $n = 2$ this gives $\mathrm{MV}(Q_1, Q_2) = \mathrm{Vol}(Q_1 + Q_2) - \mathrm{Vol}(Q_1) - \mathrm{Vol}(Q_2)$.

Exercise 7.1.3. Prove both formulae for the mixed volume from Proposition 7.1.2, in the case $n = 2$, using Definition 7.1.1. You may start by proving that $\mathrm{Vol}(\lambda_1 Q_1 + \lambda_2 Q_2)$ lies in $\mathbb{Z}[\lambda_1, \lambda_2]$ and prove the first part of Proposition 7.1.2. Then prove the second part of the proposition for $n = 2$.

One may verify that mixed volume scales in the same way as the number of common roots of a well-constrained polynomial system with generic coefficients. In particular, when some Newton polytope is expressed as a Minkowski sum, this means that the corresponding polynomial equals the product of two polynomials $f_i f_i'$. So, the mixed volume can be written as a sum of mixed volumes, which corresponds to the fact that the generic number of common roots is given by a sum of root counts, each count corresponding to a system of polynomials including either f_i or f_i'.

Such properties were used by Kushnirenko in proving a restricted version of the following theorem, for the unmixed case [Kus75]. Then, Bernstein (also spelled Bernshteĭn) stated, in [Ber75], the now-famous generalization, also known as the Bernstein-Kushnirenko-Khovanskii (BKK) bound. We are now ready to state a slight generalization of this theorem.

Theorem 7.1.4. *Given system (7.1), the cardinality of common isolated zeros in $(\overline{K}^*)^n$, counting multiplicities, is bounded by* $\mathrm{MV}(Q_1, \ldots, Q_n)$*, regardless of the dimension of the variety. Equality holds when a certain subset of the coefficients corresponding to the vertices of the Q_i's are generic.*

Newton polytopes provide a "sparse" counterpart of total degree. The same holds for mixed volume vis-à-vis Bézout's bound, which is equal to the product of all total degrees. The two bounds coincide for completely dense polynomials, because each Newton polytope is an n-dimensional unit simplex scaled by $\deg f_i$. By definition, the mixed volume of the dense system is

$$\mathrm{MV}((\deg f_1)S, \ldots, (\deg f_n)S) = \prod_{i=1}^{n} \deg f_i \;\; \mathrm{MV}(S, \ldots, S) = \prod_{i=1}^{n} \deg f_i,$$

where S is the unit simplex in $\mathbb{R}^n$ with vertex set $\{(0,\ldots,0),(1,0,\ldots,0),\ldots,(0,\ldots,0,1)\}$.

There is an intermediate bound between the classical Bézout bound and mixed volume. It is called the m-homogeneous or, simply, m-Bézout bound, and holds for multihomogeneous polynomials. Suppose that the n variables are partitioned into $r \geq 1$ sets of n_j variables each, for $j = 1,\ldots,r$. Then, $n_1 + \cdots + n_r = n$. We may assume that there is a homogenizing variable for each variable subset j such that polynomial f_i becomes homogeneous with respect to each subset, and has degree d_{ij} for $i = 1,\ldots,n$ and $j = 1,\ldots,r$. Then, the m-Bézout number is given by

$$\text{the coefficient of } \prod_{j=1}^{r} x_j^{n_j} \text{ in polynomial } \prod_{i=1}^{n} \left(\sum_{j=1}^{r} d_{ij} x_j \right).$$

This number lies always between the classical Bézout bound and the mixed volume. For a general discussion see [MSW95].

Exercise 7.1.5 (combinatorial). If all d_{ij} are equal to d_j then recover the classical Bézout's bound. Furthermore, show that the mixed volume of a system of multihomogeneous polynomials is given by the m-Bézout bound. For this, write every Newton polytope as $Q_i = \sum_j d_{ij} S_j$, where S_j is the unit simplex in n_j dimensions.

Mixed volume is usually significantly smaller than Bézout's bound for systems encountered in engineering applications. One example is the simple and generalized eigenproblems on $k \times k$ matrices. By adding an equation to ensure unit length of vectors, the Bézout bound in both cases is 2^{k+1}, whereas the number of right eigenvector and eigenvalue pairs is $2k$. This is precisely the mixed volume. We might, alternatively, employ the m-Bézout bound to the $k \times k$ system and obtain the exact count, namely k.

It is possible to generalize the notion of mixed volume to that of *stable mixed volume,* thus extending the bound to affine roots [HS97b].

The mixed volume computation is tantamount to enumerating all *mixed cells* in a mixed (tight coherent) subdivision of $Q_1 + \cdots + Q_n$. The term "decomposition" is also used in the literature, instead of "subdivision". We express the operation of Minkowski addition on n polytopes as a many-to-one function from $(\mathbb{R}^n)^n$ onto $\mathbb{R}^n$:

$$(Q_1,\ldots,Q_n) \to \sum_{i=1}^{n} Q_i \ : \ (p_1,\ldots,p_n) \mapsto \sum_{i=1}^{n} p_i.$$

To define an inverse function, i.e., a unique tuple for every point in the sum, *lifting* is a standard geometric method. Select n generic linear lifting forms $l_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1,\ldots,n$. Then define the lifted polytopes

$$\widehat{Q}_i = \{(p_i, l_i(p_i)) : p_i \in Q_i\} \subset \mathbb{R}^{n+1}, \qquad i = 1,\ldots,n.$$

Now consider the Minkowski sum $\widehat{Q}_1 + \cdots + \widehat{Q}_n$, which is a convex polytope in $\mathbb{R}^{n+1}$. The lower hull of this Minkowski sum is an n-dimensional (convex) polyhedral complex, i.e. a family of convex faces of varying dimensions that includes all subfaces, such that the intersection of any two faces is itself a face of both intersecting faces. The lower hull is defined with respect to the unit vector along the x_{n+1}-axis: It is equal to the union of all n-dimensional faces, or facets, whose inner normal vector has positive last component.

Each facet of $\sum_{i=1}^n \widehat{Q}_i$ can be written itself as a Minkowski sum $\sum_{i=1}^n \widehat{F}_i$ where every $\widehat{F}_i$ is a face of $\widehat{Q}_i,\ i = 1, \ldots, n$. The genericity of the l_i ensures two things: First, that the lower hull projects bijectively onto the Minkowski sum $\sum_{i=1}^n Q_i$ of the original polytopes. Second, it guarantees tightness, which is the formal term for expressing the fact that every lower hull facet is a unique sum of faces $\widehat{F}_i$ so that $\sum_{i=1}^n \dim \widehat{F}_i$ equals the dimension of the facet, namely n. Note that for an arbitrary lifting we would have $\sum_{i=1}^n \dim \widehat{F}_i \geq n$, but tightness means that equality holds.

The subdivision of the lower hull into faces of dimensions from 0 to n induces a subdivision of the Minkowski sum $\sum_{i=1}^n Q_i$ into cells of respective dimensions. Such a subdivision is called regular and is defined by projecting each lower-hull face onto one cell. In particular facets, whose dimension is n, are projected onto n-dimensional (hence, maximal) cells. Furthermore, each (maximal) cell σ is expressed as the Minkowski sum of faces from the Q_i: Each Minkowski sum

$$\sigma = F_1 + \cdots + F_n$$

is unique, where each F_i is a face of Q_i, so that $\sum_i \dim F_i = \dim \sigma$. Each F_i corresponds to $\widehat{F}_i$ that appears in the unique sum defining the corresponding lower-hull facet that projects onto σ. This sum is said to be optimal since it minimizes the aggregate lifting function over the given cell.

The regularity of the subdivision implies its coherence, i.e., a continuous change of the optimal expressions of every cell σ as a sum of faces. This cell complex is, therefore, a tight coherent *mixed subdivision*. We define the *mixed cells* to be precisely those where all summand faces are one-dimensional.

Proposition 7.1.6. *The mixed volume equals the sum of the volumes of all mixed cells in the mixed subdivision.*

Example 7.1.7. Consider the system

$$f_1 = c_{10} + c_{11}x_1x_2 + c_{12}x_1^2x_2 + c_{13}x_1,\ f_2 = c_{20} + c_{21}x_2 + c_{22}x_1x_2 + c_{23}x_1.$$

These polynomials have Newton polytopes and Minkowski sum as shown in Figure 7.1. The shown subdivision is achieved with $l_1 = -x_1 - 2x_2, l_2 = 4x_1 + x_2$.

It is clear that the mixed volume equals 3, which is the exact number of common roots for two generic polynomials with these supports. However, the system's Bézout number equals 4.

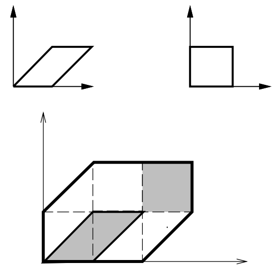


Fig. 7.1. The Newton polytopes and mixed subdivision in Example 7.1.7.

In the sequel, we shall see more examples of mixed subdivision. Some of the simplest instances appear in Examples 7.2.2 and 7.4.5.

Exercise 7.1.8. Compute the mixed volume of

$$A_1 = \{(0,0),(1,0),(2,0)\},\ A_2 = \{(0,0),(0,1),(0,2)\}.$$

Can you find a linear lifting that yields a single mixed cell, so that the mixed volume equals the volume of a single cell?

In terms of complexity classes, the computation of mixed volume is #P-complete. This computation identifies the integer points comprising a monomial basis of the quotient ring of the ideal defined by the input polynomials. Mixed, or stable mixed, cells also correspond to start systems (of binomial equations, hence with an immediate solution) for a *toric homotopy* to the original system's roots. Such issues go beyond the scope of this chapter; see Chapter 8 or [GLW99, Li97, VG95].

7.1.1 The toric resultant

For a more general introduction to resultants, one may consult Sections 1.3 and 1.6 of Chapter 1, Section 2.3 of Chapter 2, or Chapter 3. The *resultant* of a polynomial system of $n+1$ polynomials with indeterminate coefficients in n variables is a polynomial in these indeterminates, whose vanishing provides a necessary and sufficient condition for the existence of common roots of the system. Simple examples and a formal definition follow.

The resultant can be expressed by Poisson's formula, namely $C\ \prod_\alpha f_0(\alpha)$, where f_0 is one of the polynomials, evaluated at all common roots α of the other n equations, and C is a function of the coefficients of these n polynomials. It is then easy to see that the resultant is homogeneous in the coefficients of each polynomial.

The history of resultants (and elimination theory) includes such luminaries as Euler, Bézout, Cayley, and Macaulay. Different resultants exist depending on the space of the roots we wish to characterize, namely projective, affine, toric or residual [BEM01, CLO98, EM99b, Stu02]. Projective resultants (also known as classical) were historically the first to be studied and characterize the existence of projective roots. We shall focus on toric resultants below. Residual resultants were more recently introduced in order to study roots in the difference of two varieties.

Example 7.1.9. The bilinear system $f_i = c_{i0}+c_{i1}x_1+c_{i2}x_2+c_{i3}x_1x_2$, $i = 0, 1, 2$ is used in modelling a bilinear surface in $\mathbb{R}^3$ as the set of values $(f_0, f_1, f_2) \in \mathbb{R}^3$; see Figure 7.2.

Fig. 7.2. A bilinear surface patch.

The bivariate system of the f_i's has toric resultant equal to

$$\mathrm{Res} = \det \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} & 0 & 0 \\ c_{10} & c_{11} & c_{12} & c_{13} & 0 & 0 \\ c_{20} & c_{21} & c_{22} & c_{23} & 0 & 0 \\ 0 & c_{00} & 0 & c_{02} & c_{01} & c_{03} \\ 0 & c_{10} & 0 & c_{12} & c_{11} & c_{13} \\ 0 & c_{20} & 0 & c_{22} & c_{21} & c_{23} \end{bmatrix},$$

assuming the matrix $[c_{ij}]_{i,j\geq 0}$ is regular. Notice that the first three matrix rows correspond to the input polynomials, whereas the last three rows correspond to the same polynomials multiplied by x_1. This determinant has degree 2 per polynomial, which is precisely the mixed volume of two input polynomials; remark that this is the generic number of roots. Hence the determinant equals the toric resultant.

In the following sections, we shall discuss ways to construct this matrix and, ultimately, the resultant. Two alternative ways are presented in Chapter 1.

If our only tool were the projective (classical) resultant, one would consider 3 bivariate polynomials, each of total degree 2. The resultant has degree 4 per polynomial, hence 12 in total in the c_{ij}'s. For the bilinear system, certain coefficients must be specialized to zero. One can show that the projective (classical) resultant vanishes identically in this case.

The simplest case, where the classical projective and toric resultants coincide, is that of a linear system of $n+1$ equations in n variables. The determinant of the coefficient matrix is the system's resultant and, under the assumption on the non-vanishing of certain minors, it becomes zero exactly when there is a common root. Due to the linearity of the equations, this root is then unique.

Exercise 7.1.10. Using linear algebra, prove that the resultant of a linear system vanishes precisely when there exists a unique common root, provided that certain minors are nonzero. Moreover, apply Cramer's rule in order to compute each coordinate of this root as a ratio of determinants.

The question of whether two polynomials $f_1(x), f_2(x) \in K[x]$ have a common root leads to a condition that has to be satisfied by the coefficients of both polynomials; again classical and toric resultants coincide. The system's *Sylvester matrix* is of dimension $\deg f_1 + \deg f_2$ and its determinant is the system's resultant, provided the leading coefficients are nonzero. This matrix rows contain the coefficient vectors of polynomials $x^k f_j$, for $k = 0, \ldots, \deg f_i - 1$ and $\{i, j\} = \{1, 2\}$.

Bézout developed a method for computing the resultant as a determinant of a matrix of dimension equal to $\max\{\deg f_1, \deg f_2\}$. Its construction goes beyond the scope of this chapter; the reader may refer to Chapters 1 and 3.

For an illustration, consider $f_1 = a_{d_1} x^{d_1} + \cdots + a_0, f_2 = b_{d_2} x^{d_2} + \cdots + b_0$, with all coefficients nonzero. Their resultant is the determinant of the Sylvester matrix, namely

$$\begin{bmatrix} a_{d_1} & a_{d_1-1} & \cdots & a_0 & 0 & \cdots & & 0 \\ 0 & a_{d_1} & a_{d_1-1} & \cdots & a_0 & 0 & \cdots & 0 \\ \vdots & & \ddots & & & \ddots & & \\ 0 & & & a_{d_1} & a_{d_1-1} & \cdots & & a_0 \\ b_{d_2} & b_{d_2-1} & \cdots & & b_0 & 0 & \cdots & 0 \\ 0 & b_{d_2} & b_{d_2-1} & \cdots & & b_0 & 0 & 0 \\ \vdots & & \ddots & & & \ddots & & \\ 0 & & & b_{d_2} & b_{d_2-1} & \cdots & & b_0 \end{bmatrix}.$$

The interested reader may refer to Section 1.3 of Chapter 1 for a more detailed discussion on resultants of univariate polynomials.

Exercise 7.1.11. Using the greatest common divisor of f_1, f_2 prove that the resultant of these two polynomials vanishes precisely when they have a common root. Can you compute the coordinates of this root from the kernel vectors of the Sylvester matrix?

Toric resultants express the existence of toric roots. Formally,

$$f_0, \dots, f_n \in K[x_1^{\pm 1}, \dots, x_n^{\pm 1}], \tag{7.2}$$

f_i corresponding to generic point $c_i = (c_{i1}, \dots, c_{im_i})$ in the space of polynomials with support A_i. This space is identified with projective space $\mathbb{P}_K^{m_i - 1}$. Then system (7.2) can be thought of as point $c = (c_0, \dots, c_n)$. Let Z denote the Zariski closure, in the product of projective spaces, of the set of all c such that the system has a solution in $(\overline{K}^*)^n$. Note that Z is an irreducible variety.

A technical assumption is that, without loss of generality, the affine lattice generated by $\sum_{i=1}^{n+1} A_i$ is n-dimensional. This lattice is identified with $\mathbb{Z}^n$ possibly after a change of variables, which can be implemented by computing the appropriate Smith's Normal form [Stu94a].

Definition 7.1.12. *The* toric (or sparse) resultant $\mathrm{Res} = \mathrm{Res}(A_0, \dots, A_n)$ *of system (7.2) is a polynomial in* $\mathbb{Z}[c]$. *If codim*$(Z) = 1$ *then* Res *is the defining irreducible polynomial of the hypersurface* Z. *If codim*$(Z) > 1$ *then* $\mathrm{Res} = 1$.

An additional assumption we make is that the family $A_0, \dots, A_n$ is *essential*. This means that, for every proper index subset $I \subset \{0, \dots, n\}$ with cardinality $|I|$, the following holds for the dimension of certain Minkowski sums:

$$\dim \left(\sum_{i \in I} A_i \right) \geq |I|.$$

Essential support families are also discussed in Section 1.6 of Chapter 1.

Then, the toric resultant $\mathrm{Res}(A_0, \dots, A_n)$ is homogeneous in the coefficients of f_i with $\deg_{f_i} \mathrm{Res}(A_i) = \mathrm{MV}_{-i}$. The vanishing of $\mathrm{Res}(A_0, \dots, A_n)$ is a necessary and sufficient condition for the existence of roots in the projective toric variety X, corresponding to the Minkowski sum of the $n+1$ Newton polytopes. A projective toric variety is the closure of the image of the following map of the torus:

$$(\mathbb{C}^*)^n \to \mathbb{P}^m : t \mapsto \left(t^{b_0} : \cdots : t^{b_m} \right),$$

where the $b_i \in \mathbb{Z}^n$ are the vertices of the Minkowski sum. If all Newton polytopes are identical, then these are simply the vertices of the Newton polytope. For instance, when this polytope is the unit simplex, the toric resultant coincides with $\mathbb{P}^n$. In the case of bilinear systems (see Example 7.1.9), $X = \mathbb{P}^1 \times \mathbb{P}^1$. Toric varieties are also discussed in Chapter 3 as well as in [Cox95, GKZ94, KSZ92].

Some fundamental properties of the toric resultant are as follows.

- The toric resultant subsumes the classical resultant in the sense that they coincide if the polynomials are dense.
- Just as in the classical case, when all coefficients are generic, the resultant is irreducible.

- While the classical resultant is invariant under linear transformations of the variables, the toric resultant is invariant under transformations that preserve the polynomial support.
- In the case of non-generic coefficients, certain divisibility properties hold. In particular, when a system of polynomials lies in the ideal generated by another system, then the latter resultant is divisible by the former resultant.

7.2 Matrix formulae

Different means of expressing each resultant are possible, distinguished into Sylvester, Bézout and hybrid-type formulae [BEM01, CLO98, DE03, EM99b, Stu02]. Ideally, we wish to express it as a matrix determinant, a quotient of two determinants, or a divisor of a determinant where the quotient is a nontrivial extraneous factor. This section discusses matrix formulae for the toric resultant known as *toric resultant matrices.*

We restrict ourselves to Sylvester-type matrices; such matrices for the toric resultant are also known as *Newton matrices* because they depend on the input Newton polytopes. Sylvester-type matrices generalize the coefficient matrix of a linear system and Macaulay's matrix. The latter extends Sylvester's construction to arbitrary systems of homogeneous polynomials, and its determinant is a nontrivial multiple of the projective resultant. Other types of resultant matrices are discussed in Chapter 3.

The transpose of a Sylvester-type matrix corresponds to the following linear transformation:

$$(g_0, \ldots, g_n) \quad \mapsto \quad \sum_{i=0}^{n} g_i f_i, \tag{7.3}$$

where the support of each polynomial g_i is related to the matrix. If we expressed the g_i's in the monomial basis, then $(g_0, \ldots, g_n)$ would be a vector that multiplies from the left the transposed matrix (or from the right, the resultant matrix itself). The support of each g_i is the set of monomials multiplying f_i in order to define the rows that correspond to f_i. These rows contain shifted copies of the f_i coefficients. The shift is performed in such a way so as to obtain $g_i f_i$ as the product of g_i-block of the vector, multiplied by the block of rows corresponding to f_i. The reader should consult the examples of resultant matrices given above as well as in the sequel.

Overall, each row expresses the product of a monomial with an input polynomial; its entries are coefficients of that product, each corresponding to the monomial indexing the corresponding column. The degree of $\det M$ in the coefficients of f_i equals the number of rows with coefficients of f_i. This must be greater than or equal to $\deg_{f_i} \mathrm{Res}$. It is possible to pick any one polynomial so that there is an optimal number of rows containing its coefficients; this number is obviously $\deg_{f_i}$ Res. This is true both in the case of Macaulay's matrix and in the case of the Newton matrix constructions below.

7.2.1 Subdivision-based construction

There are two main approaches to construct a well-defined, square, generically nonsingular matrix M, such that Res $\mid \det M$. The second algorithm is incremental and shall be presented later. The first approach (cf. [CE93, CE00, CP93, Stu94a]), relies on a mixed (tight coherent) subdivision of the Minkowski sum

$$Q = Q_0 + \cdots + Q_n,$$

which generalizes the discussion of Section 7.1. It uses $n+1$ generic linear lifting forms $l_i : \mathbb{R}^n \to \mathbb{R}$ to define the lifted polytopes. Maximal cells in the subdivision are written uniquely as $\sigma = F_0 + \cdots + F_n$, where $F_i \subset Q_i$ and $\sum_i \dim F_i = n$. Therefore, at least one face is a vertex. The *mixed cells* are precisely those where all other summand faces are one-dimensional. If this is a vertex from Q_i, then the cell is said to be i-mixed.

It can been shown [Emi96] that the i-mixed cells are the same as the mixed cells in the mixed subdivision the n Newton polytopes $Q_0, \ldots, Q_{i-1}, Q_{i+1}, \ldots,$ Q_n, provided that we use the same lifting functions in both cases. A direct consequence is that the mixed volume of $f_0, \ldots, f_{i-1}, f_{i+1}, \ldots, f_n$ is given by the sum of volumes of all i-mixed cells, thus extending Proposition 7.1.6.

The matrix construction algorithm uses a subset of $(Q+\delta) \cap \mathbb{Z}^n$ to index the rows and columns of resultant matrix M, where $\delta \in \mathbb{R}^n$ is an arbitrarily small and *sufficiently generic* vector. This vector must perturb all integer points indexing some row (or column) of the matrix in the strict interior of a maximal cell. It can be chosen randomly and the validity of our choice can be confirmed by the matrix construction algorithm. The probability of error for a vector with uniformly distributed entries is bounded in [CE00].

Now consider an integer point p, such that $p+\delta$ lies in an arbitrary maximal cell σ. The algorithm associates to p the pair (i, j) if and only if $a_{ij} \in Q_i$ is a vertex in the optimal sum of σ and i is the maximum index of any vertex summand. The row of M corresponding to p shall contain the coefficients of polynomial

$$x^{p-a_{ij}} f_i.$$

The entries corresponding to column monomials that do not explicitly appear in the row polynomial are set to zero. If σ is i-mixed, then a_{ij} is the unique vertex summand. For non-mixed cells, the Minkowski sum has more than one vertices, and the above rule defines a matrix with the minimum number of rows with f_0, because in these cases it shall avoid the 0 index.

Therefore, the number of f_0 rows equals the number of integer points in 0-mixed cells, which equals

$$\mathrm{MV}(f_1, \ldots, f_n) = \deg_{f_0} \mathrm{Res}(A_i).$$

As for the number of f_i rows, for $i > 0$, this is larger or equal to the number of integer points in i-mixed cells. The above argument tells us that this is at least as large as $\deg_{f_i}$ Res. Now recall that the degree of the matrix determinant in the coefficients of f_i equals the number of its rows containing shifted copies of the coefficient vector of f_i. The algorithm may use an analogous rule to avoid index i if we wish the matrix to have the minimum number of rows containing f_i, for $i > 0$.

It can be proven that every principal minor of matrix M, including its determinant, is nonzero when the polynomials have generic coefficients [CE00]. The proof of this theorem uses an adequate specialization of the input coefficients, in terms of a new parameter t. In particular, the coefficient in f_i that multiplies the monomial x^{a_j} is specialized to $t^{l_i(a_j)}$, where l_i is the lifting applied to Q_i. Then, each row of the specialized matrix, indexed by some point p, is multiplied by the power $t^{h-l_k(a_s)}$. Here, h denotes the vertical distance of $p \in \mathbb{R}^n$ to the lower hull of $\sum_{i\geq 0} \widehat{Q}_i$ and we have assumed that p has been associated to the pair (k, s). The last step in the proof establishes that the product of all diagonal entries in the new matrix equals the trailing term of its determinant with respect to t.

Moreover, it is not so hard to show that the determinant of M vanishes whenever Res $= 0$. We thus arrive at the following theorem.

Theorem 7.2.1 ([CE93, CE00]). *We are given an overconstrained system with fixed supports. With the above notation, matrix M is well-defined and square. Its determinant is generically nonzero and divisible by the toric resultant* Res.

Example 7.2.2. Let us apply the subdivision-based algorithm to construct Sylvester's matrix. Take

$$f_0 = c_{00} + c_{01}x, \ f_1 = c_{10} + c_{11}x + c_{12}x^2.$$

There are two possible subdivisions obtained with linear liftings; one is shown in Figure 7.3, along with the δ perturbation.

For illustration, we note that the algorithm associates to point 2 the pair $(1, 2)$, i.e. the matrix row indexed by x^2 shall contain the coefficients of $x^{2-2}f_1 = f_1$. A similar argument builds the other rows of the matrix. The reader may check that this is indeed the well-known Sylvester matrix.

Example 7.2.3. For $n = 2$, let us apply the subdivision-based algorithm in the case of linear polynomials. Take

$$f_i = c_{i0} + c_{i1}x_1 + c_{i2}x_2, \ i = 0, 1, 2.$$

One possible linear lifting induces the subdivision in Figure 7.4. The same figure shows the perturbation of choice, so that we recover the matrix of the system's coefficients, as expected. In fact, any vector $\delta \in \mathbb{R}_{>0}$ would do.

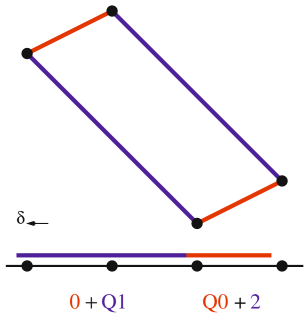


Fig. 7.3. The Minkowski sum of the lifted Newton segments and the induced subdivision in Example 7.2.2.

Then, there are three integer points in the perturbed Minkowski sum, namely $(1,2),(1,1)$, and $(2,1)$. They are associated, respectively, to pairs $[2,(0,1)]$, $[1,(0,0)]$ and $[0,(1,0)]$. For instance, the row indexed by $x_1x_2^2$ shall contain polynomial $x^{(1,2)-(0,1)}f_2 = x^{(1,1)}f_2$.

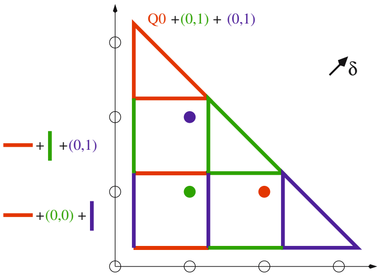


Fig. 7.4. The mixed subdivision and the perturbation with respect to the original Minkowski sum.

The resultant matrix is therefore

$$M = \begin{bmatrix} c_{01} & c_{02} & c_{03} \\ c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{bmatrix},$$

with rows corresponding to the polynomials $x_1x_2f_i$ and columns indexed by $x_1^2x_2,\ x_1x_2^2,\ x_1x_2$.

There is a greedy variant from [CP93] of the subdivision-based algorithm. It starts with a single row, corresponding to some integer point, and proceeds iteratively by adding new rows (and columns) as need be. For a given set of rows, the column set comprises all columns required to express the row polynomials. For a given set of columns, the rows are updated to correspond to the same set. The algorithm continues by adding rows and the corresponding columns until a square matrix has been obtained.

Example 7.2.4. Consider a system of 3 polynomials in 2 unknowns:

$$
\begin{aligned}
f_0 &= c_{01} + c_{02}xy + c_{03}x^2y + c_{04}x, \\
f_1 &= c_{11}y + c_{12}x^2y^2 + c_{13}x^2y + c_{14}x, \\
f_2 &= c_{21} + c_{22}y + c_{23}xy + c_{24}x.
\end{aligned}
$$

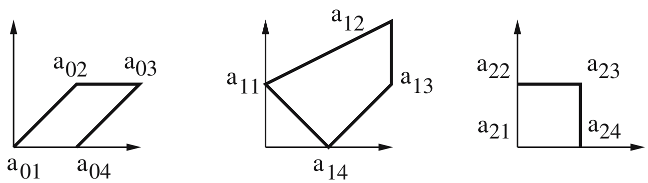
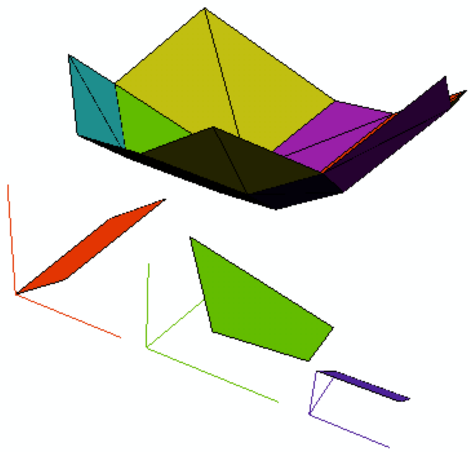


Fig. 7.5. The supports and Newton polytopes in Example 7.2.4.

The Newton polytopes are shown in Figure 7.5. The mixed volumes are $\mathrm{MV}(Q_0, Q_1) = 4$, $\mathrm{MV}(Q_1, Q_2) = 4$, $\mathrm{MV}(Q_2, Q_0) = 3$, so the toric resultant's total degree is 11. Compare this with the Bézout numbers of these subsystems: $8, 6, 12$; hence the projective resultant's total degree is 26.

Assume that the lifting functions are $l_0(x, y) = Lx + L^2y, l_1(x, y) = -L^2x - y, l_2(x, y) = x - Ly$, where $L \gg 1$. The lifted Newton polytopes and the lower hull of their Minkowski sum is shown below. These functions are sufficiently generic since they define a mixed subdivision where every cell is uniquely defined as the Minkowski sum of faces $F_i \subset Q_i$.



The lower hull of the Minkowski sum of the lifted Q_i's is then projected to the plane, yielding generically a mixed subdivision of Q. Figure 7.6 shows $Q+\delta$ and the integer points it contains; notice that every point belongs to a unique maximal cell. Every maximal cell σ is labeled by the indices of the Q_i vertex or vertices appearing in the unique Minkowski sum $\sigma = F_0 + \cdots + F_n$, with ij denoting vertex $a_{ij} \in Q_i$. For instance, point $(1,0)$ belongs to a maximal cell $\sigma = a_{01} + F + F'$, where F, F' are the edges $(a_{14}, a_{13}) \subset Q_1$ and (a_{21}, a_{24}) respectively. The corresponding row in the matrix will be filled in with the coefficient vector of $x^{(1,0)} f_0$.

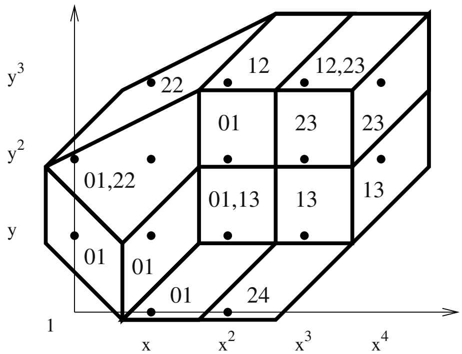


Fig. 7.6. A mixed subdivision of Q perturbed by $(-3/8, -1/8)$, in Example 7.2.4.

The Newton matrix M appears below with rows and columns indexed by the integer points in the perturbed Minkowski sum. M contains, by construction, the minimum number of f_0 rows, namely 4. The total number of rows is $4 + 4 + 7 = 15$, i.e., the determinant degree is higher than optimal by 1 and 3, respectively, in the coefficients of f_1 and f_2.

$$
\begin{array}{cc}
 & \begin{array}{ccccccccccccccc} 1,0 & 2,0 & 0,1 & 1,1 & 2,1 & 3,1 & 0,2 & 1,2 & 2,2 & 3,2 & 4,2 & 1,3 & 2,3 & 3,3 & 4,3 \end{array} \\
\begin{array}{c} 1,0 \\ 2,0 \\ 0,1 \\ 1,1 \\ 2,1 \\ 3,1 \\ 0,2 \\ 1,2 \\ 2,2 \\ 3,2 \\ 4,2 \\ 1,3 \\ 2,3 \\ 3,3 \\ 4,3 \end{array} &
\left[\begin{array}{ccccccccccccccc}
c_{01} & c_{04} & 0 & 0 & c_{02} & c_{03} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
c_{21} & c_{24} & 0 & c_{22} & c_{23} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & c_{01} & c_{04} & 0 & 0 & 0 & c_{02} & c_{03} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & c_{01} & c_{04} & 0 & 0 & 0 & c_{02} & c_{03} & 0 & 0 & 0 & 0 & 0 \\
c_{14} & 0 & c_{11} & 0 & c_{13} & 0 & 0 & 0 & c_{12} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & c_{14} & 0 & c_{11} & 0 & c_{13} & 0 & 0 & 0 & c_{12} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{22} & c_{23} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{22} & c_{23} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{01} & c_{04} & 0 & 0 & 0 & c_{02} & c_{03} \\
0 & 0 & 0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{22} & c_{23} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & c_{14} & 0 & 0 & c_{11} & 0 & c_{13} & 0 & 0 & 0 & c_{12} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{22} & c_{23} & 0 & 0 \\
0 & 0 & 0 & c_{14} & 0 & 0 & c_{11} & 0 & c_{13} & 0 & 0 & 0 & c_{12} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{22} & c_{23} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{22} & c_{23}
\end{array}\right]
\end{array}
$$

The greedy version produces a matrix with dimension 14 which can be obtained by deleting the row and the column corresponding to point $(1, 3)$.

The subdivision-based approach can be coupled with the existence of a minor in the Newton matrix that divides the determinant so as to yield the exact toric resultant [D'A02]. D'Andrea has proposed a recursive lifting procedure that gives a much lower value to a chosen vertex of Q_0. The cells whose optimal sum does not contain this vertex are then further subdivided by assigning this special role to a vertex of Q_1, and so on. This generalizes Macaulay's famous quotient formula that yields the exact projective resultant [Mac02].

The existence of a non-recursive algorithm, relying on a single lifting, is still open in the general case. It is, nonetheless, possible for $n = 2$ and for families of sufficiently different Newton polytopes. A glimpse of what this lifting may look like is offered by the hybrid matrix constructed in [DE01b].

Example 7.2.5 (Continued from Example 7.1.9). The bilinear system $f_i = c_{i0} + c_{i1}x_1 + c_{i2}x_2 + c_{i3}x_1x_2$, $i = 0, 1, 2$, despite its apparent simplicity, does not admit an optimal toric resultant matrix, when we apply the subdivision-based algorithm. In contrast, the greedy variant may yield an optimal matrix and the incremental algorithm of the next section produces the optimal 6×6 matrix in Example 7.1.9. It is possible to construct the following 9×9 numerator matrix, using the subdivision-based algorithm:

$$M = \begin{array}{c} \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} & 0 & 0 & 0 & 0 & 0 \\ c_{10} & c_{11} & c_{12} & c_{13} & 0 & 0 & 0 & 0 & 0 \\ c_{20} & c_{21} & c_{22} & c_{23} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{00} & c_{01} & c_{02} & 0 & 0 & c_{03} \\ 0 & c_{10} & 0 & c_{12} & c_{13} & 0 & c_{11} & 0 & 0 \\ 0 & 0 & c_{20} & c_{21} & 0 & c_{23} & 0 & c_{22} & 0 \\ 0 & c_{20} & 0 & c_{22} & c_{23} & 0 & c_{21} & 0 & 0 \\ 0 & 0 & c_{10} & c_{11} & 0 & c_{13} & 0 & c_{12} & 0 \\ 0 & 0 & 0 & c_{10} & c_{11} & c_{12} & 0 & 0 & c_{13} \end{bmatrix} \end{array} \quad \begin{array}{c} f_0 \\ f_1 \\ f_2 \\ x_1x_2f_0 \\ x_1f_1 \\ x_2f_2 \\ x_1f_2 \\ x_2f_1 \\ x_1x_2f_1 \end{array}$$

The choice was $\delta = (\frac{2}{3}, \frac{1}{2})$ and the lifting is such that one vertex of the first polytope has an infinitesimal lifting value compared to the other values. It is now possible to define a denominator matrix M', of dimension 3, which is a submatrix of M. It is defined by the rows indexed by polynomials $f_1, f_2, x_1x_2f_1$ and the respective columns; these correspond precisely to the integer points in non-mixed cells. The ratio of the determinants yields precisely the toric resultant.

7.2.2 Incremental construction

The second algorithm [EC95], is *incremental* and yields usually smaller matrices and, in any case, no larger than those of the subdivision algorithm. The flexibility of the construction makes it suitable for overconstrained systems. On the downside, there exists a randomized step so certain properties of the subdivision-based construction cannot be guaranteed *a priori*.

The selection of integer points, which correspond to monomials multiplying the row polynomials, uses a vector $v \in (\mathbb{Q}^*)^n$. The goal is to choose an adequate subset of integer points in

$$Q_{-i} := \sum_{j=0, j\neq i}^{n} Q_j, \; i = 0, \ldots, n.$$

This is achieved by first sorting all points $p \in Q_{-i} \cap \mathbb{Z}^n$ according to their distance, along v, from the boundary. This distance is defined as follows, for point p:

$$v\text{-distance}(p) := \max\{s \in \mathbb{R}_{\geq 0} \, : \, p + sv \in Q_{-i}\}.$$

The construction is incremental, in the sense that successively larger point sets are considered by decreasing the lower bound on the v-distance of the set's points. For given point sets, a candidate matrix is defined. If the number of rows is at least as large as the number of columns and it has full rank for generic coefficients, then the algorithm terminates and returns a nonsingular maximal square submatrix. The determinant of this submatrix is a nontrivial multiple of the toric resultant; otherwise, new rows (and columns) are added to the candidate.

In those cases where a minimum matrix of Sylvester type provably exists [SZ94, WZ94], the incremental algorithm produces this matrix. For general multi-homogeneous systems, the best vector is obtained in [DE03]. These are precisely the systems for which v can be deterministically specified; otherwise, a random v can be used. Different choices can be tried out so that the smallest matrix may be chosen.

Example 7.2.6 (Continued from Example 7.2.4). Figure 7.7 shows Q_{-0} in bold and randomly chosen vector $v = (20, 11)$. The different point subsets in Q_{-0} with respect to v-distance are shown by the thin-line polygons. In fact, the thin lines represent contours of fixed v-distance. The final point set from Q_{-0} is the following, shown with the respective v-distances:

$$\{(0,1;3/20), (1,0;1/10), (1,1;1/10), (1,2;1/11)\}.$$

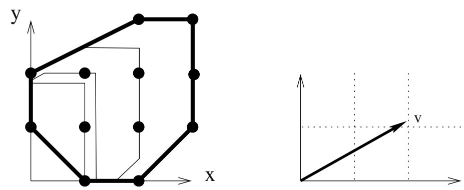


Fig. 7.7. Q_{-0} subsets with different v-distance bounds and vector v.

This v leads to a 13×12 nonsingular matrix M shown below. Deleting the last row defines the 12×12 resultant submatrix.

$$\begin{array}{cc}
 & \begin{array}{cccccccccccc} 1,2 & 2,2 & 0,1 & 1,1 & 2,1 & 3,1 & 1,0 & 2,0 & 3,2 & 2,3 & 3,3 & 0,2 \end{array} \\
\begin{array}{c} 0,1 \\ 1,0 \\ 1,1 \\ 1,2 \\ 0,0 \\ 1,0 \\ 1,1 \\ 0,1 \\ 0,1 \\ 1,1 \\ 1,0 \\ 2,1 \\ 2,2 \end{array} &
\begin{bmatrix}
c_{02} & c_{03} & c_{01} & c_{04} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & c_{02} & c_{03} & c_{01} & c_{04} & 0 & 0 & 0 & 0 \\
0 & c_{02} & 0 & c_{01} & c_{04} & 0 & 0 & 0 & c_{03} & 0 & 0 & 0 \\
c_{01} & c_{04} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{02} & c_{03} & 0 \\
0 & c_{12} & c_{11} & 0 & c_{13} & 0 & c_{14} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & c_{11} & 0 & c_{13} & 0 & c_{14} & c_{12} & 0 & 0 & 0 \\
c_{11} & 0 & 0 & 0 & c_{14} & 0 & 0 & 0 & c_{13} & 0 & c_{12} & 0 \\
0 & c_{13} & 0 & c_{14} & 0 & 0 & 0 & 0 & 0 & c_{12} & 0 & c_{11} \\
c_{23} & 0 & c_{21} & c_{24} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_{22} \\
c_{22} & c_{23} & 0 & c_{21} & c_{24} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & c_{22} & c_{23} & 0 & c_{21} & c_{24} & 0 & 0 & 0 & 0 \\
0 & c_{22} & 0 & 0 & c_{21} & c_{24} & 0 & 0 & c_{23} & 0 & 0 & 0 \\
0 & c_{21} & 0 & 0 & 0 & 0 & 0 & 0 & c_{24} & c_{22} & c_{23} & 0
\end{bmatrix}
\end{array}$$

Other techniques to reduce matrix size (and mixed volumes) include the introduction of new variables to express subexpressions which are common to several input polynomials. For an illustration, see [Emi97].

Clearly, mixed volume captures the inherent complexity of algebraic problems in the context of sparse elimination and thus provides lower bounds on the complexity of algorithms. On the other hand, several toric elimination algorithms rely on Minkowski sums of Newton polytopes. Therefore, a crucial question in deriving output-sensitive upper bounds is the relation between mixed volume and the volume of these Minkowski sums. In manipulating mixed volumes, some fundamental results can be found in [Sch93]. In particular, the Aleksandrov-Fenchel inequality leads to the following bound [Emi96, Lut86]:

$$\mathrm{MV}^n(Q_1, \ldots, Q_n) \geq (n!)^n \mathrm{Vol}(Q_1) \cdots \mathrm{Vol}(Q_n).$$

For a system of Newton polytopes Q_i, define its *scaling factor* s to be the minimum real value so that $Q_i + t_i \subset s\, Q_\mu$ for all Q_i, where Q_μ is the polytope of minimum euclidean volume and the $t_i \in \mathbb{R}^n$ are arbitrary translation vectors. Clearly, $s \geq 1$ and s is finite if and only if all polytopes have an affine span of the same dimension. Let e denote the basis of natural logarithms, and suppose that the volumes $\mathrm{Vol}(Q_i) > 0$ for all i. Then, for a well-constrained system, we have

$$\mathrm{Vol}\left(\sum_{i=1}^{n} Q_i\right) = O(e^n s^n) \mathrm{MV}(Q_1, \ldots, Q_n),$$

whereas for an overconstrained system the same techniques yield

$$\mathrm{Vol}\left(\sum_{i=0}^{n} Q_i\right) = O\left(\frac{e^n s^n}{n}\right) \sum_{i=0}^{n} \mathrm{MV}_{-i},$$

where $\mathrm{MV}_{-i} = \mathrm{MV}(Q_0, \ldots, Q_{i-1}, Q_{i+1}, \ldots, Q_n)$ [Emi96].

As a consequence, the asymptotic bit complexity of both subdivision-based and incremental algorithms is singly exponential in n, proportional to the total degree of the toric resultant, and polynomial in the number of Q_i vertices, provided all $\mathrm{MV}_{-i} > 0$.

Newton matrices, including the candidates constructed by the incremental algorithm, are characterized by a structure that generalizes the Toeplitz structure and has been called *quasi-Toeplitz* [EP02] (cf. [CKL89]). By exploiting this structure, determinant evaluation has quasi-quadratic arithmetic complexity and quasi-linear space complexity in the matrix dimension (here "quasi" means that polylogarithmic factors are ignored). The efficient implementation of this structure is open today and is important for the competitiveness of the entire approach.

7.3 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 modelling. Given are rational parametric expressions

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

over some field K of characteristic zero. The implicitization problem consists in computing the smallest algebraic hypersurface 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. Resultants offer an efficient approach for this problem, but face certain questions due to degeneracy conditions, discussed below. Several other algorithms exist for this problem, including methods based on Gröbner bases, moving surfaces, and residues. Their enumeration goes beyond the scope of this chapter; cf. also, Chapter 3.

Implicitization is equivalent to eliminating all parameters t from the polynomial system

$$f_i(t) = p_i(t) - x_i q(t), \; i = 0, \ldots, n,$$

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, provided that it does not vanish and the parametrization is generically one-to-one. Otherwise, the resultant is a power of the implicit equation. More subtle is the case where the resultant is identically zero. This happens precisely when there exist values of t, known as *base points*, for which the f_i vanish for all x_i; in other words, the $p_i(t)$ and $q(t)$ evaluate to zero. 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. But higher codimension presents a harder problem.

Besides cases where the (toric) resultant vanishes, another problem with non-generic coefficients is that the resultant matrix may be identically singular. We understand that avoiding degeneracies is an important problem, whose relevance extends beyond the question of implicitization with base points. In [DE01a], a toric (sparse) projection operator is defined by perturbing the subdivision-based matrix such that, after specialization, this operator is not identically zero but vanishes on roots in the proper components of the variety, including all isolated roots.

This is a standard idea in handling degeneracies in the case of resultants. In the classical context, Canny [Can90] perturbed each f_i by adding a new factor $\epsilon x_i^{d_i}$, where $i = 1, \ldots, n$, and f_0 by adding ϵ, where ϵ is a positive infinitesimal indeterminate. Rojas proposed a perturbation scheme for toric resultants in [Roj99a] which yields a perturbed resultant of low degree in ϵ but is, nonetheless, rather expensive to compute. Our scheme generalizes [Can90] and requires virtually no extra computation besides the matrix construction.

Suppose we have a family $p := (p_0(x) \ldots, p_n(x))$ of Laurent polynomials such that $\mathrm{supp}(p_i) \subset A_i$, and $\mathrm{Res}(p_0, \ldots, p_n) \neq 0$. The Toric Generalized Characteristic Polynomial (p-GCP) is

$$C_p(\epsilon) := \mathrm{Res}\,(f_0 - \epsilon p_0, \ldots, f_n - \epsilon p_n)\,.$$

Let $C_{p,k}(y_1, \ldots, y_m)$ be the coefficient of $C_p(\epsilon)$ of lowest degree in ϵ, namely k. The coefficient $C_{p,k}$ is a suitable *projection operator*. In fact, the polynomials p_i may have random coefficients and support including precisely those monomials of f_i which appear on the diagonal of the toric resultant matrix. The perturbation has been implemented in Maple; see also Section 7.5.

Example 7.3.1 (Continued from Example 7.2.4). 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 toric resultant vanishes for all c_{1j} 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 $\{-1\} \times \mathbb{C}$. For a specific lifting and matrix construction, the trailing coefficient in the perturbed determinant is that of ϵ^2 and equals

$$-(c_{12}c_{13})(c_{14} - c_{11} + c_{12} - c_{13})(c_{14} + c_{11} - c_{12} + c_{13}).$$

So 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.

The next example illustrates the perturbation method in applying toric resultants for system solving.

Example 7.3.2. This is the example of [Roj99a]. To the system

$$f_1 := 1+2x-2x^2y-5xy+x^2+3x^3y,\;\; f_2 := 2+6x-6x^2y-11xy+4x^2+5x^3y,$$

we add $f_0 := u_1x+u_2y+u_0$, which does not have to be perturbed. We use the function `spresultant` from Maple library MULTIRES to construct a 16×16 matrix `M` 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. Here is the Maple code for these operations, where `e` stands for ϵ:

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

For certain ω and δ, we have used $p_1 := -3x^2 + x^3y$, $p_2 := 2 + 5x^2$. The perturbed determinant has maximum and minimum degree in ϵ, respectively, 12 and 1. The trailing coefficient 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: $(-u_0 + u_1)\;(27\,u_2 + 40\,u_1 - 40\,u_0)$.

Example 7.3.3. In the robot motion planning implementation of Canny's *roadmap algorithm* in [HP00], numerous "degenerate" systems are encountered. Let us examine a 3×3 system, where we hide x_0 to obtain dense polynomials of degrees $3, 2, 1$:

$$\begin{aligned}
f_0 = {} & 54x_1{}^3 - 21.6x_1{}^2x_2 - 69.12x_1x_2{}^2 + 41.472x_2{}^3 + (50.625 + 75.45x_0)\, x_1{}^2 \\
& + (-92.25 + 32.88x_0)\, x_1x_2 + (-74.592x_0 + 41.4)\, x_2{}^2 + \\
& + (131.25 + 19.04x_0{}^2 - 168x_0)x_1 + \left(-405 + 25.728x_0{}^2 + 126.4x_0\right) x_2 + \\
& + (-108.8\, x_0{}^2 + 3.75\, x_0 + 234.375), \\
f_1 = {} & -37.725\, x_1{}^2 - 16.44\, x_1x_2 + 37.296\, x_2{}^2 + (-38.08x_0 + 84)\, x_1 + \\
& + (-63.2 - 51.456x_0)\, x_2 + (2.304x_0{}^2 + 217.6x_0 - 301.875), \\
f_2 = {} & 15\, x_1 - 12\, x_2 + 16\, x_0.
\end{aligned}$$

The Maple function `spresultant` applies an optimal perturbation to an identically singular 14×14 matrix in x_0. Now $\det M(\epsilon)$ is of degree 14 and the trailing coefficient of degree 2, which provides a bound on the number of affine roots. We obtain

$$\phi(x_0) = \left(x_0 - \frac{1434}{625}\right)\left(x_0 - \frac{12815703325}{21336}\right),$$

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

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.

Example 7.3.4. Let us consider the de-homogenized version of a system defined in [Bus01b]:

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

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

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

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 7.3.5. The problem of computing the sparse, or toric, discriminant of a polynomial specified by its support can be formulated as an implicitization problem [DS02, GKZ94]. Let us fix the polynomial support in $\mathbb{Z}^m$, and suppose that the support's cardinality equals $m+1+s,\ s \geq 0$. The case $s=2$ was studied in [DS02] and reduces to curve implicitization, though the approach used in that article was not based on implicitization.

Here $s=3$, so we have a surface implicitization problem with base points. 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 $x_0, \ldots, x_{s-1}$.

The parametric expressions for the x_i's and the ensuing implicitization problem shall be defined in terms of the entries of some matrix B, specified from the support of the input polynomial. Its row dimension is s and its column dimension equals the cardinality of the polynomial support. We do not go into the technical details of deriving B from the support.

Let us consider a specific example with $m=3$ and $s=3$, hence the support cardinality equals 7. The problem reduces to implicitizing the parametric surface given by

$$x_i = \prod_{j=1}^{7} \left(b_{0j} + t_1 b_{1j} + t_2 b_{2j}\right)^{b_{ij}}, \quad i = 0, 1, 2,$$

where the matrix $B = (b_{ij})$, for $i = 0, \ldots, 2,\ j = 1, \ldots, 7$, is as follows:

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

There are base points forming components of codimension 2, including a single affine base point $(1, -1)$. Our algorithm constructs a 33×33 matrix, whose perturbed determinant has a trailing term of degree 3 in ϵ. The corresponding coefficient has total degree 14 in x_0, x_1, x_2. When factorized, it yields the precise implicit equation, which is of degree 9 in x_0, x_1, x_2.

7.4 Implicit support

In this section, we exploit information on the support of the toric resultant in order to predict the support of the implicit equation of a parametric (hyper)surface.

Our approach is to consider the extreme monomials i.e., the vertices of the Newton polytope of the toric resultant Res. The output support scales with the sparseness of the parametric polynomials and is much tighter than the one predicted by degree arguments. In many cases, we obtain the exact support of the implicit equation, as seen by applying our Maple program. Moreover, it is possible to specify certain coefficients in this equation. Our motivation comes mainly from two implicitization algorithms which apply interpolation, namely the direct method of [CGKW01] and the one based on perturbations (cf. Section 7.3 or [MC92]).

The initial form $\mathrm{In}_\omega(F)$ of a multivariate polynomial F in k variables, with respect to some functional $\omega : \mathbb{Z}^k \to \mathbb{R}$, is the sum of all terms in F which maximize the inner product of ω with the corresponding exponent vector. Let us define

$$k := |A_0| + \cdots + |A_n|,$$

then ω defines a *lifting* function on the input system, by lifting every support point $a \in A_i$ to $(a, \omega(a)) \in \mathbb{Z}^n \times \mathbb{R}$. This generalizes the linear lifting of Section 7.2. The lower hull facets of the lifted Minkowski sum correspond to maximal cells of an induced coherent mixed subdivision of Q. If ω is sufficiently generic, then this subdivision is tight; in the sequel, we assume our mixed subdivision is both coherent and tight and denote it by Δ_ω. If $F_i \in A_i$ is a vertex summand of an i-mixed cell, then the corresponding coefficient in f_i is denoted by c_{iF_i}. We recall our assumption that the A_i span $\mathbb{Z}^n$.

Theorem 7.4.1. *The initial form of the toric resultant* Res *with respect to a generic ω equals the monomial*

$$In_\omega(\mathrm{Res}) = \prod_{i=0}^{n} \prod_{F} c_{iF_i}^{\mathrm{Vol}(F)}, \tag{7.4}$$

where $\mathrm{Vol}(\cdot)$ *denotes ordinary Euclidean volume and the second product is over all mixed cells of type i in the mixed subdivision Δ_ω.*

For a detailed proof of this theorem, see [Stu94a]. This proof can be obtained from the toric resultant matrix construction, by means of the subdivision-based algorithm. Let us use the same specialization of the coefficients in terms of a new parameter t, as in the discussion that leads to Theorem 7.2.1. Then, the resultant becomes univariate in t and the proof is completed by relating, on the one hand, the degree of $\mathrm{In}_\omega(\mathrm{Res})$ in t and, on the other, the sum of all exponents in expression (7.4). The latter, for fixed i, equals $\mathrm{MV}_{-i} = \deg_{f_i} \mathrm{Res}$.

For a generic vector ω, the initial form $\mathrm{In}_\omega(\mathrm{Res})$ corresponds to a vertex of the Newton polytope of the resultant Res. It is precisely the vertex with inner normal ω. So, by varying the lifting ω, we can compute all vertices of this Newton polytope, hence a superset of the resultant's support.

A bijective correspondence exists between the extreme monomials and the configurations of the mixed cells of the A_i. So, it suffices to compute all distinct mixed-cell configurations, as discussed in [MC00, MV99].

Another (simpler) means of reducing the number of relevant mixed subdivisions is by bounding the number of cells. This bound is usually straightforward to compute in small dimensions (e.g. when $n = 2, 3$) and reduces drastically the set of mixed subdivisions. For instance, when studying the implicitization of a biquadratic surface, the total number of mixed subdivisions is 19728, whereas those with 8 cells is 62.

In certain special cases, we can be more specific about the Newton polytope of the toric resultant. First, its dimension equals $k-2n-1$ [GKZ94, Stu94a]. Certain corollaries follow: For essential support families (defined in [Stu94a]), a 1-dimensional Newton polytope of Res is possible if and only if all polynomials are binomials. The only resultant polytope of dimension 2 is the triangle; in this case the support cardinalities must be 2 and 3. For dimension 3, the possible polytopes are the tetrahedron, the square-based pyramid, and polytope $N_{2,2}$ given in [Stu94a]; the support cardinalities are respectively $2,2$ and 3.

One corollary of Theorem 7.4.1 (and of its proof) is that the coefficients of all extreme monomials are in $\{-1,1\}$ [GKZ91, CE00, Stu94a]. Sturmfels [Stu94a] also specifies, for all extreme monomials, a way to compute their precise coefficients. But this requires computing several coherent mixed subdivisions, and goes beyond the scope of the present chapter.

The so-called *Cayley trick* introduces a new point set $C := \{(z, a_{0j}, 1) \ : \ a_{0j} \in A_0\} \cup \{(e_i, a_{ij}, 1) \ : \ i = 1, \dots, n, \ a_{ij} \in A_i\} \subset \mathbb{Z}^{2n+1}$, where $z = (0, \dots, 0) \in \mathbb{N}^n$ is the zero vector and $e_i = (0, \dots, 0, 1, 0, \dots, 0) \in \mathbb{N}^n$ has a unit at the i-th position and $n-1$ zeroes.

Theorem 7.4.2. *The problem of computing all mixed subdivisions of supports $A_0, \dots, A_n$, which lie in $\mathbb{Z}^n$, is equivalent to computing all regular triangulations of the set C defined above. This set contains $k_0 + \cdots + k_n$ points, where $k_i = |A_i|$.*

Example 7.4.3 (Continued from Example 7.2.2). The Cayley trick in the univariate case goes as follows. Consider $f_0 = c_{00} + c_{01}x$, $f_1 = c_{10} + c_{12}x^2$, then the points in the set C appear in the columns of matrix

$$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 2 \end{bmatrix}.$$

There are two possible triangulations of these points, namely

$$\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right), \ \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right),$$

which is the one shown in Figure 7.3, and

$$\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right), \ \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right).$$

Efficient algorithms (and implementations) exist for computing all regular triangulations of a point set [Ram01]. Regular are those triangulations that can be obtained by projection of a lifted triangulation.

We produce a superset of the monomials in the support of the implicit equation of the input. Consider, as in Section 7.3 the polynomials $f_i(t) = p_i(t) - x_i q(t)$, where we ignore the specific values of the coefficients.

This is an interesting feature of the algorithm, namely that it considers the monomials in the parametric equations but not their actual coefficients. This shows that the algorithm is suitable for use as a preprocessing off-line step in CAGD computations, where one needs to compute thousands of examples with the same support structure in real time. This handles the implicitization of (multiparametric) families of (hyper)surfaces, indexed by one or more parameters.

Of course, the generic resultant coefficients are eventually specialized to functions of the x_i. Then, any bounds on the implicit degree in the x_i may be applied, in order to reduce the final support set. One step yields as by-product all partial mixed volumes MV_{-i} for $i = 0, \ldots, n$, and hence the implicit degree separately in the x_i variables.

We examine our method on some small examples, and summarize the results in Table 7.1 below.

Example 7.4.4. We consider the Folium of Descartes, shown also in Figure 7.8. $x = 3t^2/(t^3+1),\ y = 3t/(t^3+1)$.

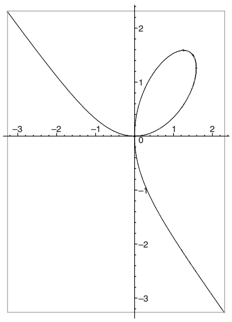


Fig. 7.8. The Folium of Descartes

The output monomials are $\{y^3, x^3, x^3\, y^3, x\, y, y^2\, x^2\}$. After applying the degree bound $d = 3$ we obtain the support $\{y^3, x^3, x\, y\}$, which is optimal, since the implicit equation is $x^3 + y^3 - 3\, x\, y = 0$.

Example 7.4.5. An example in 3 dimensions comes from [Buc88b]; the surface is drawn in Figure 7.9. The parametric expressions are: $x = s\ t,\ y = s\ t^2,\ z = s^2$.

In order to apply toric elimination theory, we consider polynomials

$$f_0 = c_{00} - c_{01}st,\ f_1 = c_{10} - c_{11}st^2,\ f_2 = c_{20} - c_{21}s^2.$$

There are the following two possible mixed subdivisions, each containing exactly three maximal cells, all of which are mixed, see Figure 7.10.

The computed support is optimal and the implicit equation is $x^4 - y^2 z = 0$.

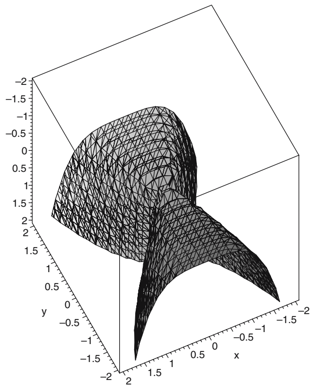


Fig. 7.9. The surface in Example 7.4.5.

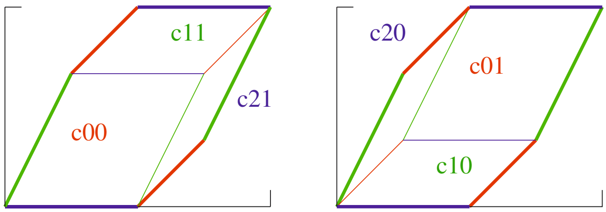


Fig. 7.10. Mixed cells in the subdivisions, with vertex summands shown.

Example 7.4.6. Let us consider a system attributed to Fröberg and discussed in Chapter 1.

$$x = t^{48} - t^{56} - t^{60} - t^{62} - t^{63}, \ y = t^{32}.$$

The Minkowski sum is the segment $Q_0 + Q_1 = [0, 95]$. One type of triangulations, obtained from a non-linear lifting, divides it to the following 3 cells (which are all segments):

$$(Q_0' + 0),\ (a + Q_1),\ (Q_0'' + 32), \ \text{ where } Q_0' = [0, a],\ Q_0'' = [a, 63],$$

and $a \in A_0 = \{0, 48, 56, 60, 62, 63\}$. Every such triangulation yields a support point y^a. The triangulation $(0+Q_1), (Q_0+32)$, which is induced from a linear lifting, yields support point x^{32}. Note that only certain of these monomials are extreme when we consider the resultant in terms of all input coefficients, in order for the respective coefficients to lie in $\{-1, 1\}$.

Therefore, we find, as the toric resultant support, the triangle with vertices $(32, 0), (0, 48)$ and $(0, 63)$. Equivalently, it is delimited by the y-axis and the lines $y = -(3/2)x + 48$ and $y = -(63/32)x + 63$, as shown in Figure 7.11.

Counting the points with integer coordinates inside (and on the sides) of the triangle, we see that there are 257 such points, which is seen to be optimal by actually computing the resultant.

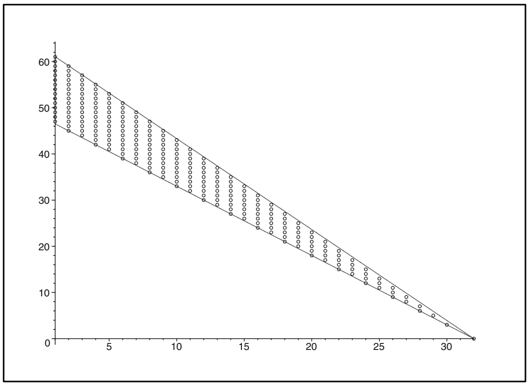


Fig. 7.11. Toric resultant support.

Table 7.1. Predicting the implicit support.

Problem	Input Degree	Degree of Implicit Eq.	General # monomials	# monomials from [EK03]
Unit Circle	2	2	6	3
Descartes Folium, Ex. 7.4.4	3	3	10	3
Fröberg-Dickenstein, Ex. 7.4.6	63	63	1057	257
Buchberger, Example 7.4.5	1,2	4	35	2
Busé, Example 7.3.4	3	5	56	4
Bilinear, Example 7.1.9	1,1	2	10	9

Example 7.4.7. The well-known bicubic surface represents a challenge for our current implementation: $x = 3\,t\,(t-1)^2 + (s-1)^3 + 3\,s$, $y = 3\,s\,(s-1)^2 + t^3 + 3\,t$, $z = -3s(s^2-5s+5)t^3\ -3(s^3+6s^2-9s+1)t^2 +\ t(6s^3+9s^2-18s+3) - 3s(s-1)$. We computed 737129 regular triangulations (by TOPCOM) [Ram01]. For illustration purposes, we show one of them:

```
{2,3,4,7,13},{3,4,5,7,13},{3,5,6,7,13},{3,6,9,13,14},
{6,9,12,13,14},{3,6,9,14,15},{6,9,12,14,15},{6,12,13,14,16},
{6,12,14,15,16},{6,12,15,16,17},{3,6,9,15,18},{6,9,12,15,18},
{6,12,15,17,18},{3,9,15,18,19},{3,6,9,18,19},{6,9,12,18,19},
{6,12,16,17,20},{6,12,17,18,20},{3,6,9,19,23},{6,9,12,19,23},
{6,12,19,22,23},{6,12,22,23,24},{6,12,23,24,25},{3,6,9,23,26},
{6,9,12,23,26},{6,12,23,25,26},{0,2,4,7,13},{3,6,7,9,13},
{6,12,18,19,22},{6,12,18,20,24},{6,7,9,12,13},{6,12,18,22,24}.
```

The size of the file is 383 MBytes. This underlines the fact that we should not compute all regular triangulations but only the mixed-cell configurations.

7.5 Algebraic solving by linear algebra

To solve well-constrained system (7.1) by the resultant method we define an overconstrained system and apply the resultant matrix construction. For a more comprehensive discussion the reader may refer to Chapters 2 and 3, or [CLO98, EM99c].

One advantage of resultant-based methods is that resultant matrix M need be computed only once, for all systems with the same supports. So this step is thought of as being carried out off-line, while the matrix operations to approximate all isolated roots for each coefficient specialization constitute the online part. Numerical issues for the latter are discussed in [Emi97, EM99c].

Resultant matrices reduce system solving to certain standard operations in computer algebra. In particular, univariate or multivariate determinants can be computed by evaluation and interpolation techniques. However, the determinant development in the monomial basis may be avoided because there are algorithms for univariate polynomial solving as well as multivariate polynomial factorization which require only the values of these polynomials at specific points; cf. e.g. [Pan97]. All of these evaluations would exploit the quasi-Toeplitz structure of Sylvester-type matrices [CKL89, EP02].

We present two ways of defining an overconstrained system. The first method adds to the given system *an extra polynomial*, namely

$$f_0 = u_0 + u_1x_1 + \cdots + u_nx_n \ \in (K[u_0,\ldots,u_n])[x_1^{\pm 1},\ldots,x_n^{\pm 1}],$$

thus yielding a well-studied object, the u-resultant. Coefficients $u_1,\ldots,u_n$ may be randomly specialized or left as indeterminates; in the latter case, solving reduces to factorizing the u-polynomial. It is known that the u-resultant factorizes into linear factors $u_0 + u_1\alpha_1 + \cdots + u_n\alpha_n$ where $(\alpha_1,\ldots,\alpha_n)$ is an isolated root of the original system. This is an instance of Poisson's formula. Now, u_0 is usually an indeterminate that we shall denote by x_0 below for uniformity of notation. Matrix M will describe the multiplication map for f_0 in the coordinate ring of the ideal defined by the system in (7.1).

An alternative way to obtain an overconstrained system is by *hiding* one of the original variables in the coefficient field and consider the system as follows (we modify the previous notation to unify the subsequent discussion):

$$f_0,\ldots,f_n \in (K[x_0])\,[x_1^{\pm 1},\ldots,x_n^{\pm 1}].$$

M is a matrix polynomial in x_0, and may not be linear.

An important issue concerns the degeneracy of the input coefficients. This may result in the trivial vanishing of the toric resultant or of $\det M$ when there is an infinite number of common roots (in the torus or at toric infinity) or simply due to the matrix constructed. An infinitesimal perturbation has been proposed [DE01a] which respects the structure of Newton polytopes and is computed at no extra asymptotic cost, cf. Section 7.3.

The perturbed determinant is a polynomial in the perturbation variable, whose leading coefficient is nonzero whereas the least significant coefficient is $\det M$. Irrespective of which coefficients vanish, there is always a trailing nonzero coefficient which vanishes when x_0 takes its values at the system's isolated roots, even in the presence of positive-dimensional components. This univariate polynomial is known as a *projection operator* because it projects the roots to the x_0-coordinate. Univariate polynomial solving thus yields these coordinates. Again, the u-resultant allows us to recover all coordinates via multivariate factoring.

A basic property of resultant matrices is that right vector multiplication expresses evaluation of the row polynomials. Specifically, multiplying by a column vector containing the values of column monomials q at some $\alpha \in (\overline{K}^*)^n$ produces the values of the row polynomials

$$\alpha^p f_{i_p}(\alpha).$$

Computationally it is preferable to have to deal with as small a matrix as possible. To this end we partition M into four blocks M_{ij} so that the upper left submatrix M_{11} is square, independent of x_0, and of maximal dimension so that it remains well-conditioned.

If the matrix is obtained from the subdivision-based algorithm, then we know that M_{11} corresponds to the integer points in the 0-mixed cells. More precisely, the columns of M_{11} are indexed by those points, whereas its rows contain the multiples of f_0 with the corresponding monomials. It can be proven that these monomials form a basis of the quotient ring defined by the ideal of $f_1, \ldots, f_n$, namely $K[x_1^{\pm 1}, \ldots, x_n^{\pm 1}]/\langle f_1, \ldots, f_n \rangle$. For a proof, see [Emi96, PS96].

Once M_{11} is specified, let $A(x_0) = M_{22}(x_0) - M_{21}(x_0) M_{11}^{-1} M_{12}(x_0)$. To avoid computing M_{11}^{-1}, we may use its LU (or QR) decomposition to solve $M_{11} X = M_{12}$ and compute $A = M_{22} - M_{21} X$.

Let $\mathcal{E}$ be the monomial set indexing the rows and columns of M and let $B \subset \mathcal{E}$ index A. If $(\alpha_0, \alpha) \in \overline{K}^{n+1}$ is a common root with $\alpha \in \overline{K}^n$, then $\det A(\alpha_0) = 0$ and, for any vector $v' = [\cdots \alpha^q \cdots]$, where q ranges over B, $A(\alpha_0) v' = 0$. Moreover,

$$\begin{bmatrix} M_{11} & M_{12}(\alpha_0) \\ 0 & A(\alpha_0) \end{bmatrix} \begin{bmatrix} v \\ v' \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow M_{11} v + M_{12}(\alpha_0) v' = 0,$$

determines v once v' has been computed. Vector $[v, v']$ contains the values of every monomial in $\mathcal{E}$ at α.

It can be shown that $\mathcal{E}$ affinely spans $\mathbb{Z}^n$ and an affinely independent subset can be computed in polynomial time [Emi96]. Given v, v' and these points, we can compute the coordinates of α. If all independent points are in B then v' suffices for solving. To find the vector entries that will allow us to recover the root coordinates, it is typically sufficient to search in B for pairs of entries corresponding to q_1, q_2 such that $q_1 - q_2 = (0, \ldots, 0, 1, 0, \ldots, 0)$. This lets us compute the i-th coordinate, if the unit appears at the i-th position. In general, the problem of choosing the best vector entries for computing the root coordinates is open, and different choices may lead to different accuracy.

To reduce the problem to an eigendecomposition, let r be the dimension of $A(x_0)$, and $d \geq 1$ the highest degree of x_0 in any entry. We wish to find all values of x_0 at which

$$A(x_0) = x_0^d A_d + x_0^{d-1} A_{d-1} + \cdots + x_0 A_1 + A_0$$

becomes singular. These are the eigenvalues of the *matrix polynomial*. Furthermore, for every eigenvalue λ, there is a basis of the kernel of $A(\lambda)$ defined by the right eigenvectors of the matrix polynomial associated to λ. If A_d is nonsingular then the eigenvalues and right eigenvectors of $A(x_0)$ are the eigenvalues and right eigenvectors of monic matrix polynomial $A_d^{-1} A(x_0)$. This is always the case when adding an extra linear polynomial, since $d = 1$ and $A_1 = I$ is the $r \times r$ identity matrix; then

$$A(x_0) = -A_1(-A_1^{-1} A_0 - x_0 I).$$

Generally, the *companion matrix* of a monic matrix polynomial is a square matrix C of dimension rd. The eigenvalues of C are precisely the eigenvalues λ of $A_d^{-1} A(x_0)$, whereas its right eigenvector $w = [v_1, \ldots, v_d]$ contains a right eigenvector v_1 of $A_d^{-1} A(x_0)$ and $v_i = \lambda^{i-1} v_1$, for $i = 2, \ldots, d$.

We now address the question of a singular A_d. The following *rank balancing* transformation in general improves the conditioning of A_d. If matrix polynomial $A(x_0)$ is not identically singular for all x_0, then there exists a transformation $x_0 \mapsto (t_1 y + t_2)/(t_3 y + t_4)$ for some $t_i \in \mathbb{Z}$, that produces a new matrix polynomial of the same degree and with nonsingular leading coefficient. If A_d is ill-conditioned for all linear rank balancing transformations, then we build the matrix pencil and apply a *generalized eigendecomposition* to solve $C_1 x + C_0$. This returns pairs (α, β) such that matrix $C_1 \alpha + C_0 \beta$ is singular with an associated right eigenvector.

Acknowledgment

I acknowledge financial support by INRIA Sophia-Antipolis (France) through the bilateral collaboration "Calamata" between the Galaad Group and the Department of Informatics & Telecommunications of the National Kapodistrian University of Athens.