On the Support of the Implicit Equation of Rational Parametric Hypersurfaces

Ioannis Z. Emiris [1], Ilias S. Kotsireas [2]

Abstract. We propose the use of various tools from algebraic geometry, with an emphasis on toric (or sparse) elimination theory, in order to predict the support of the implicit equation of a parametric hypersurface. The problem of implicitization lies at the heart of several algorithms in computer-aided design and geometric modeling, two of which are immediately improved by our contributions. We believe that other methods of implicitization shall be able to benefit from our work. In particular we use, on the one hand, degree bounds, formulated in terms of the mixed volume of Newton polytopes and on the other, information on the support of the toric (or sparse) resultant. In many cases, we obtain the exact support of the implicit equation.

1 Introduction

In this paper we apply several tools from algebraic geometry, with an emphasis on toric (or sparse) elimination theory, in order to predict the support of the implicit equation of a parametric hypersurface. The problem of switching from a rational parametric representation to an implicit, or algebraic, representation of a curve, surface, or hypersurface lies at the heart of several algorithms in computer-aided design and geometric modeling.

In particular we use, on the one hand, bounds on the total degree of the implicit expression, as well as bounds on its degree in each variable. For tightness, we formulate these bounds in terms of Newton polytopes and mixed volumes, which exploit any structure in the parametric equations. On the other hand, we exploit information on the support of the toric (or sparse) resultant by considering the extreme monomials as described in [GKZ94,Stu94]. In many cases, we obtain the exact support of the implicit equation.

Our motivation comes mainly from two implicitization algorithms. The first is [CGKW01], where the authors propose a new method for implicitization of parametric families of curves, surfaces and hypersurfaces, using essentially linear algebra. The method has a very wide range of applicability, can handle base points, and works both symbolically and numerically, depending on the way one performs the integrations. It may be improved, both theoretically and in what regards the implementation as follows: First, it looks for an implicit equation of a particular degree at a time. This implies that any information on the degree of the implicit equation (such as upper bounds) may accelerate execution.

More importantly, the method constructs a symmetric singular square matrix and computes a basis of its nullspace. The dimension of this matrix equals the number of possible monomials in the implicit equation, since the rows and columns are indexed by these monomials. Without any constraints on the set of monomials, the dimension is $\binom{m+n}{m}$, where the number of parametric equations is n and the algorithm seeks an implicit equation of total degree m. The worked out examples show that we succeed in obtaining substantial efficiency improvements, because we constrain in advance the monomials that will appear in the implicit equation, hence diminishing dramatically the size of the matrices involved as illustrated in the table below.

Our second motivation are algorithms based on perturbed resultant matrices, which yield the implicit equation even in the presence of base points, e.g. [DE01,MC92]. The problem reduces to sparse interpolation, which is substantially accelerated when we can accurately predict the output support. More specifically, the algorithm of Ben-Or and Tiwari requires a number of evaluations which is linear in the bound on the support cardinality; cf. [BOT88,Zip93].

The comparative table below shows a synopsis of the results of the execution of the algorithm on some examples (see section 7 for all the details). We refer to our method as IPSOS.

[1] Ontario Research Centre for Computer Algebra (ORCCA), University of Western Ontario, London, N6A 5B7 ON, Canada, emiris@scl.csd.uwo.ca. Permanent affiliation: INRIA Sophia-Antipolis, France, and Department of Informatics & Telecommunications, University of Athens, Greece.

[2] Ontario Research Centre for Computer Algebra, University of Western Ontario, London N6A 5B7, ON, Canada Ilias.Kotsireas@orcca.on.ca, and Department of Computing, Wilfrid Laurier University, Waterloo ON, Canada.

Curve / Surface	parametric (bi)degree	implicit degree	General # monomials	# monomials from IPSOS
Unit Circle	2	2	6	3 (optimal)
Descartes Folium	3	3	10	3 (optimal)
Buchberger	1,2	4	35	2 (optimal)
Busé	3	5	56	4 (optimal)
Bilinear	1,1	2	10	9 (optimal)

This paper is structured as follows: In section 2 we introduce some algebraic tools that we will use in the sequel, in particular toric elimination theory with an emphasis on the toric resultant. In section 3 we summarize classical algebraic geometry arguments that allow us to predict the degree of the implicit equation in advance. We apply these arguments to predict the degrees of the implicit equation in each variable separately. In section 4 we describe the algorithm for predicting the support of the implicit equation by combining the tools mentioned in the previous sections. In section 6 we present several examples of implicitization of curves and surfaces; the computed support turns out to be optimal in all of these examples. Some of them are fully worked out to give the implicit equation by applying the method from [CGKW01]. In section 5 we give some details on our Maple implementation of the algorithm and the interoperability of the Maple code with other public-domain C/C++ stand-alone programs that we use, mainly for performing computations with polyhedra. Finally, in section 7 we provide some conclusions and some ideas for future work.

2 Toric elimination theory

This section overviews our algebraic tools, coming for the most part from the theory of toric (or sparse) elimination. For a more comprehensive discussion, the reader may consult e.g. [CLO98,GKZ94].

Let $A_i \subset \mathbb{Z}^n$ be a finite set and consider generic Laurent polynomials (i.e. with integer exponents) in n variables $x = (x_1, \ldots, x_n)$:

$$f_i(x) = \sum_{a \in \overset{\circ}{A}_i} c_{ia}\ x^a, \quad c_{ia} \neq 0.$$

Then A_i is the *support* of polynomial f_i and its *Newton polytope* $Q_i \subset \mathbb{R}^n$ is the convex hull of the support A_i. The following bound is also known as the Bernstein-Khovanskii-Kushnirenko (BKK) number.

Theorem 1. *The mixed volume $MV(Q_1, \ldots, Q_n)$ of the Newton polytopes $Q_1, \ldots, Q_n \subset \mathbb{R}^n$ corresponding to polynomials $f_1, \ldots, f_n \in K[x_1^{\pm 1}, \ldots, x_n^{\pm 1}]$ bounds the degree of the toric variety of these polynomials.*

Mixed volume generalizes Bézout's classical bound given by the product of total degrees in the sense that it reduces to Bézout's bound for dense polynomials but is in general tighter. The underlying toric variety is embedded in a projective space of high dimension and contains $(\overline{K}^*)^n$ as a dense subset, where $\overline{K}$ is the algebraic closure of the coefficient field K and $\overline{K}^* = \overline{K} \setminus \{0\}$.

The mixed volume can be computed by means of a mixed subsivision; mixed subsivisions are discussed below. Once such a subsivision is computed, the sum of volumes of all mixed cells equals the mixed volume.

Consider an over-constrained system of polynomials $f_0, \ldots, f_n \in K[x_1^{\pm 1}, \ldots, x_n^{\pm 1}]$, with respective supports $A_0, \ldots, A_n \subset \mathbb{Z}^n$.

Definition 1. *The* toric (or sparse) resultant *$R(f_0, \ldots, f_n)$ is a polynomial in $\mathbb{Z}[c_{ia}]$, homogeneous in the coefficients of each f_i, with degree equal to $MV_{-i} := MV(Q_0, \ldots, Q_{i-1}, Q_{i+1}, \ldots, Q_n)$. The resultant vanishes after a specialization of the coefficients iff the specialized system of $f_0, \ldots, f_n$ has a solution in the toric variety associated to the Newton polytopes of the f_i.*

The toric resultant is also known as the toric (or sparse) *mixed* resultant in order to emphasize the fact that the supports A_i may be different; when all A_i are identical, the system is unmixed.

2.1 Extreme monomials of the toric resultant

Certain works, including [GKZ94,Stu94] have studied the Newton polytope of R, thus providing a point set containing its support. In particular, [GKZ94] describes certain homogeneities of R with respect to c, which would yield a description of the hyperplanes defining the facets of the Newton polytope of R. But this study is not extended to toric resultants in general dimension over arbitrary input supports. An alternative approach is to specify the vertices of this Newton polytope, i.e. the extreme monomials which appear in the toric resultant. In [Stu94], the author shows that the extreme monomials correspond to the image of a many-to-one map, and indicates how to compute them. A more theoretical discussion of certain of these facts is found in [GKZ94, Sect. 8.3].

For these results, we need some concepts from polyhedral combinatorics. Consider any collection $\{A_i\}_{i\in I}$ of supports, for some set $I \subset \{0,\dots,n\}$. Its rank, denoted $\mathrm{rk}(I)$, is the rank of the affine lattice generated by the $\sum_{i\in I} A_i$. A collection $\{A_i\}_{i\in I}$, for some I, is *essential* iff $\mathrm{rk}(I) = |I|-1$ and $\mathrm{rk}(J) \geq |J|$ for every proper subset $J \subset I$. Remark that an essential collection contains no singleton. The initial form $init_\omega(f)$ 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 by the corresponding exponent vector.

When $k = |A_0| + \cdots + |A_n|$, then ω defines a *lifting* function on the input system, by lifting every support point $a \in A_i$ to $(a, \omega(a)) \in \mathbb{Z}^n \times \mathbb{R}$. The lifted supports, denoted by $\widehat{Q}_i$, lie in $\mathbb{R}^{n+1}$. Their Minkowski sum is $\widehat{Q} = \widehat{Q}_1 + \cdots + \widehat{Q}_n$; its lower hull projects bijectively, along the last coordinate, to $Q = Q_1 + \cdots + Q_n \subset \mathbb{R}^n$. The lower hull facets then correspond to maximal *cells* of an induced *coherent mixed decomposition* of Q.

If ω is sufficiently generic, then this decomposition is *tight*; in the sequel, we assume our mixed decompositions are both coherent and tight and denote it by Δ_ω. Then, maximal cells of the form $F = F_0 + \cdots + F_n$, where $\dim F_j = 1$ for all $j \in \{0,\dots,n\}$ except for one value, are called *mixed*. It is clear that the $(n+1)$st summand must have dimension 0; if this is the i-th summand, then the cell is said to be of type i or i-mixed. An important remark, to be applied later, is that the sum of volumes of all i-mixed cells equals the partial mixed volume MV_{-i}, for any $i \in \{0,\dots,n\}$.

The corresponding coefficient in f_i is denoted by c_{iF_i}; its monomial is x^{F_i} and $F_i \in A_i$.

Theorem 2 ([Stu94]). *Suppose that $\{A_0,\dots,A_n\}$ is essential. Then the initial form of the toric (mixed) resultant R with respect to a generic ω equals the monomial*

$$init_\omega(R) = \prod_{i=0}^{n} \prod_{F} c_{iF_i}^{vol(F)}$$

where $vol(\cdot)$ denotes ordinary Euclidean volume and the second product is over all mixed cells of type i of the tight mixed coherent decomposition Δ_ω.

It is clear that 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 [MV99].

Another (simpler) means of reducing the number of relevant mixed decompositions is by restricting attention to those with a specific number of cells. This number is usually straightforward to compute in small dimensions (e.g. when $n = 1, 2$, as in the imlicitization of curves and surfaces) and reduces drastically the set of mixed decompositions. For instance, when studying the implicitization of a biquadratic surface, the total number of mixed decompositions 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 $k - 2n - 1$, where $k = |A_0| + \cdots + |A_n|$ is the sum of the support cardinalities [GKZ94,Stu94]. Certain corollaries follow: For essential support families, a 1-dimensional Newton polytope of R is possible iff 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 [Stu94]; the support cardinalities are respectively $2, 2$ and 3.

It is known that the coefficients of all extreme monomials are in $\{-1, 1\}$ [GKZ91,CE00,Stu94]. This information may be used for numeric purposes. Sturmfels [Stu94] also specifies, for the extreme monomials, a way to compute the precise coefficients. But this requires computing several coherent mixed decompositions, and goes beyond the scope of the present report.

2.2 The Cayley trick

For background information and proofs see [GKZ94,MV99,Stu94]. Here we briefly indicate the elements which are important to our approach.

Theorem 3. *There exists a transformation, known as the Cayley trick, that presents the problem of computing all mixed decompositions of $A_0, \ldots, A_n \subset \mathbb{Z}^n$ as an equivalent problem of computing all regular triangulations of a set of $|A_0| + \cdots + |A_n|$ points in $\mathbb{Z}^{2n+1}$. This point set corresponds to the columns of the matrix below, where $k_i = |A_i|$.*

Let us describe the point set used in the Cayley trick. Let $k_0, \ldots, k_n \in \mathbb{N}$ denote the cardinalities of the input supports A_i. Then, it suffices to consider the point set corresponding to the column vectors of the following matrix, namely the points $(e_i, a_{ij}) \in \mathbb{Z}^{2n+1}$ for $i = 0, \ldots, n$, where $e_i = (0, \ldots, 0, 1, 0, \ldots, 0) \in \mathbb{N}^{n+1}$ has a unique unit at the i-th position and another n zeroes:

$$\begin{pmatrix} \overbrace{1\,1\ldots 1}^{k_0} & 0\ldots 0 & \cdots & 0\ldots 0 \\ 0\ldots 0 & \overbrace{1\,1\ldots 1}^{k_1} & 0\ldots 0 & 0\ldots 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0\ldots 0 & \cdots & 0\ldots 0 & \overbrace{1\,1\ldots 1}^{k_n} \\ a_{01}\ldots a_{0k_0} & a_{11}\ldots a_{1k_1} & \ddots & a_{n1}\ldots a_{nk_n} \end{pmatrix}$$

Efficient algorithms exist for computing all regular triangulations of a point set [DL00,Ram02]. These algorithms typically require that the input points be given in homogeneous coordinates, so we add a last row of units in the matrix above. In this case, we can omit the first row of the matrix above; see the examples for an illustration.

3 Degree bounds for the implicit equation

In this section we sketch some techniques to predict the total degree of the implicit equation, given the rational parametric equations. In conjunction with standard algebraic geometry arguments, (see e.g. [CLO97]) we use toric elimination theory to exploit any sparseness in the input equations. In addition, we adapt the approach to compute the degree of the implicit equation in each variable separately.

For the sake of simplicity, we shall describe our arguments in affine space instead of projective space, but this is no lack of generality. In addition, we motivate the discussion using a plane curve with rational parametric equations as follows:

$$x_0 = P_0(t_1)/Q(t_1), \quad x_1 = P_1(t_1)/Q(t_1) \tag{1}$$

where $P_0(t_1), P_1(t_1), Q(t_1)$ are univariate polynomials in t_1.

Total degree When we intersect a plane curve with a generic straight line, we obtain generically[3] a certain number of points. This number corresponds to the degree of the implicit equation. To carry out this computation algebraically, form an equation of a generic straight line $a\ x_0 + b\ x_1 + c = 0$, substitute x_0 and x_1 by (1), clear out denominators and compute the degree of the resulting equation in t_1. This degree will be the total degree of the implicit equation.

In general, we intersect the parametric hypersurface with n generic linear equations in $x_0, \ldots, x_n$, where n denotes the number of parameters; above we had $n = 1$. The parametric expressions are substituted in these n equations to yield a non-linear system of n polynomials in n indeterminates, namely the parameters $t_1, \ldots, t_n$. The degree of the (toric) variety defined by this system is bounded by the corresponding mixed volume, which thus bounds the total degree of the implicit equation. Remark that the support of each polynomial in t is the union of the supports of the $x_iQ(t) - P_i(t)$, seen as polynomials in t. Therefore all equations have the same support and the same Newton polytope, hence the mixed volume equals $n!$ multiplied by the volume of this Newton polytope.

[3] The term "generic" here is taken to mean that the line is sufficiently random, for instance it cannot be a tangent line to the curve

Degree in each variable The above argument can be adapted to compute the degree in x_0, x_1 of the implicit equation of the curve separately. To compute the degree of the implicit equation in x_0, intersect the curve with a straight line parallel to the x_0 axis, namely $x_1 = K_1$ where K_1 is a generic constant. Then, as before, x_1 is substituted by its parametric expression in (1). The degree of the resulting equation in t_1 bounds the degree of the implicit equation in x_0.

In general, we would like to bound the degree of the implicit equation in some variable x_j, $j \in \{0, \ldots, n\}$. Then, intersect the implicit equation by the set of generic linear equations $x_i = K_i$ for $j \neq i \in \{0, \ldots, n\}$ and replace the x_i by their parametric expressions. The resulting system of n equations in n indeterminates $t_1, \ldots, t_n$ is well-constrained, and its mixed volume bounds the number of its isolated roots. Hence, this mixed volume bounds the implicit degree in x_i. This is simply the mixed volume of the polynomials $x_i Q(t) - P_i(t)$, $i \neq j$, seen as polynomials in the parameters t.

4 Implicitization with Polynomial Sparse Optimized Support (IPSOS)

In this section we describe our algorithm, named IPSOS, which allows one to estimate the support of the implicit equation of a curve, surface or hypersurface. This information can subsequently be used by implicitization algorithms such as those in [CGKW01,DE01,MC92]. For general information on implicitization, the reader may consult [CLO97,MC92].

Our algorithm applies to rational parametric (hyper)surfaces. The main idea is that, given the parametric expressions $x_i = P_i(t)/Q(t)$, for $i = 0, \ldots, n$, we regard them as polynomials $f_i = x_i Q(t) - P_i(t)$ in the parameters $t = (t_1, \ldots, t_n)$. Then, the implicitization problem is equivalent to eliminating the parameters t; the implicit equation equals the resultant of the f_i, provided there are no base point (i.e. singularities) and that the parametrization is one-to-one (i.e. proper). If the latter condition is violated, then the resultant gives us a multiple of the implicit equation. For simplicity, we may assyme the given parametrization is one-to-one.

Our toric elimination tools shall be applied to the polynomials f_i, 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 implicitization of families of (hyper)surfaces is the so-called *generic implicitization*.

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 are applied, in order to reduce the support set which is output.

`INPUT:` Rational Parametric Equations of a Hypersurface

$$x_0 = P_0(t)/Q(t), \ldots, x_n = P_n(t)/Q(t) \tag{2}$$

where $t = (t_1, \ldots, t_n)$ and $\gcd(P_i(t), Q(t)) = 1, i = 0, \ldots, n$.

`OUTPUT:` A superset of the monomials in the support of the implicit equation of (2).

1. Define the polynomials $f_i = x_i Q(t) - P_i(t), i = 0, \ldots, n$ and look at them as polynomials in t: $f_i = \sum_{\alpha_{ij} \in A_i} c_{ij} t^{\alpha_{ij}}$, where $A_i \subset \mathbb{Z}^n$.
2. Apply the Cayley trick to construct a matrix described above, then compute all regular triangulations of the corresponding point set, which yield all mixed subdivisions of $A_0 + \cdots + A_n$.
3. Obtain the extreme monomials of the Newton polytope of the resultant from the mixed subdivisions. Then compute the support of the resultant.
4. Transform the support, which is a set of monomials of the form $\prod c_{ij}^{e_{ij}}$, to a set of monomials in the $x_0, \ldots, x_n$.
5. Use the implicit degree bound to eliminate any of the monomials computed at the previous step that cannot appear in the implicit equation, i.e. whose degree is higher than the total implicit degree.

Step 2 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.

Step 3 may be analyzed into certain substeps, in different ways. Given the vertices of a polytope in dimension $k = k_0 + \cdots + k_n$, there are algorithms for computing all integer points in its interior. Some efficient implementations though require that the polytope be described in terms of its facets, so we may have to produce these facets from the set of extreme points.

Notice that in the last step, we may also use implicit degree bounds in each separate variable x_i. But these bounds should already be taken into account when the algorithm computes the toric resultant support. The latter claim follows from toric resultant theory and the homogeneities of the toric resultant as a polynomial in the coefficients of the f_i.

5 Implementation of the Algorithm

A preliminary implementation of the algorithm in Maple 8 will soon be available upon request from the authors. The name of the package is IPSOS(**I**mplicitization with **P**olynomial **S**parse **O**ptimized **S**upport). Besides Maple 8 functions, it makes use of certain Linux/Unix commands as well as publicly available software for Linux/Unix. The following programs were actually used during the development stages of the algorithm. Most of them are required to be locally installed for our Maple 8 overall routine to run.

- The C Program PORTA [CLS99] developed by Thomas Christof and Andreas Loebel is a collection of routines for analyzing polytopes and polyhedra, computing the facet presentation of a polytope given its vertices, as well as for enumerating all integral points inside a polytope (though the latter does not seem to be a fast algorithm).
- The Maple V program PUNTOS [DL00] developed by Jesús A. De Loera allows us to compute regular triangulations of point sets. Its applicability is limited by reasons of efficiency as well as by the fact that it cannot well handle very small examples. But a more robust alternative (albeit not in Maple) is TOPCOM below.
- The C++ program TOPCOM version 0.11.1 [Ram02] was developed by Jörg Rambau for regular triangulations of point sets. In using it, an important option is to specify the number of simplices in the triangulations of interest. We also experimented with symmetries, which do not seem to accelerate execution.
- The C program Mixvol, which is an implementation of the incremental mixed volume algorithm from [EC95].

Some possibilities for improvement are the following: First, recall that a bijective correspondence exists between the extreme monomials and the mixed subdivisions of the A_i, which is studied in [MV99]. Unfortunately, we were unable to find an implementation of this work, hence it is not yet used in our software.

Another possibility of improvement is to use the library PolyLib [Wil02]. The Polyhedral Library (PolyLib for short) operates on objects made up of unions of polyhedra of any dimension. In particular, it may be able to compute integer points in polyhedra of high dimensions.

6 Examples

In this section we present a number of examples of implicitization of curves and surfaces given by rational function parameterizations, using the techniques developed above for predicting the degree and the support. Each example serves to illustrate different aspects of the endeavor.

6.1 Unit Circle

Suppose that we are given the following rational parameterization of the unit circle:

$$x = \frac{t^2 - 1}{t^2 + 1}, \quad y = \frac{2\,t}{t^2 + 1}. \tag{3}$$

First we write the equations as polynomials in t: $x\,t^2 + x - t^2, y\,t^2 + y - 2\,t$. The Cayley trick constructs the matrix

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 2 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

and the corresponding coefficients are $\{x+1, x-1\}, \{y, -2, y\}$ in the f_i. The corresponding 5 regular triangulations are:

$\{\{1,2,3\},\{2,3,4\},\{2,4,5\}\}, \{\{1,2,3\},\{2,3,5\}\}, \{\{1,3,4\},\{1,2,4\},\{2,4,5\}\}, \{\{1,3,4\},\{1,4,5\},\{1,2,5\}\}, \{\{1,2,5\},\{1,3,5\}\},$

which yield the following extreme monomials: $[0,2,2,0,0], [1,1,0,2,0], [2,0,0,0,2]$.

The enumeration of all support points gives: $[2,0,0,0,2], [0,2,2,0,0], [1,1,1,0,1], [1,1,0,2,0]$. Then, the candidate monomials are: $\{1, y^2, x^2, y^2\,x, y^2\,x^2\}$. Using the degree argument ($d=2$ in this case) we obtain the following monomials

$$\{1, y^2, x^2\}$$

This result is optimal and can be used to write down the actual implicit equation of the unit circle using one of the implicitization methods. The implicitization method described in [CGKW01] employs, in principle, a 6×6 symmetric singular matrix to solve this problem. Using the sparse support information we obtained from the algorithm, we reduce the size of the problem to a 3×3 matrix. We start with the vector $v = [1, x^2, y^2]$ and construct the 3×3 matrix M below. Substituting the parametric forms of x, y from 3 and integrating for $t \in [-1,1]$ we obtain matrix G:

$$M = \begin{bmatrix} 1 & x^2 & y^2 \\ x^2 & x^4 & x^2y^2 \\ y^2 & x^2y^2 & y^4 \end{bmatrix} \qquad G = \begin{bmatrix} 2 & -\pi+4 & -2+\pi \\ -\pi+4 & 16/3-3/2\,\pi & 1/2\,\pi-4/3 \\ -2+\pi & 1/2\,\pi-4/3 & 1/2\,\pi-2/3 \end{bmatrix}$$

whose nullspace is spanned by the vector $[-1,1,1]$. This vector multiplied by v gives the implicit equation of the unit circle, namely: $-1+x^2+y^2=0$.

6.2 Folium of Descartes

This example is taken from [CLO97].

$$x = \frac{3\,t^2}{t^3+1}, \quad y = \frac{3\,t}{t^3+1}. \tag{4}$$

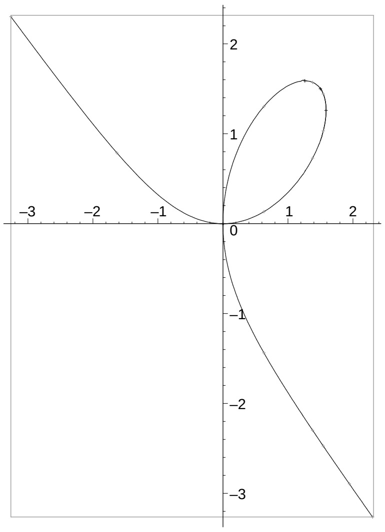

Fig. 1. Folium of Descartes

The candidate monomials are

$$\{y^3, x^3, x^3\,y^3, x\,y, y^2\,x^2\}.$$

After intersecting with the degree bound which is $d = 3$ we obtain the monomials $\{y^3, x^3, x\,y\}$.

Using the implicitization method of [CGKW01], this example requires, in principle, a 10×10 symmetric singular matrix. Using the sparse support information obtained by the algorithm, we construct the 3×3 symmetric singular matrix

$$\begin{bmatrix} x^2y^2 & x^4y & xy^4 \\ x^4y & x^6 & x^3y^3 \\ xy^4 & x^3y^3 & y^6 \end{bmatrix}$$

and performing the substitutions and the integrations for $t \in [0, \ldots, 1]$ we obtain the matrix

$$\begin{bmatrix} -\frac{13}{8} + 4/3 \ln(2) + 4/9\sqrt{3}\pi & -\frac{263}{64} + 7/3 \ln(2) + \frac{7}{9}\sqrt{3}\pi & -\frac{49}{64} + 5/3 \ln(2) + 5/9\sqrt{3}\pi \\ -\frac{263}{64} + 7/3 \ln(2) + \frac{7}{9}\sqrt{3}\pi & -\frac{779}{80} + 14/3 \ln(2) + \frac{14}{9}\sqrt{3}\pi & -\frac{829}{320} + 7/3 \ln(2) + \frac{7}{9}\sqrt{3}\pi \\ -\frac{49}{64} + 5/3 \ln(2) + 5/9\sqrt{3}\pi & -\frac{829}{320} + 7/3 \ln(2) + \frac{7}{9}\sqrt{3}\pi & \frac{47}{160} + \frac{8}{9}\sqrt{3}\pi + 8/3 \ln(2) \end{bmatrix}$$

whose nullspace is of dimension 1 and is generated by the vector $[-3, 1, 1]$. This shows that the implicit equation of the Descartes's Folium is:

$$x^3 + y^3 - 3\,x\,y = 0.$$

Since there are no zero entries in the nullvector, we see that all of the monomials predicted by our algorithm (and only these) appear in the implicit equation. This means that the result of the algorithm is optimal.

6.3 Example from [Buc88]

$$x = rt,\ y = rt^2,\ z = r^2.$$

The implicitization method described in [CGKW01] employs, *a priori*, a 35×35 symmetric singular matrix to solve this problem. Using the degree bound and the sparse support information we can use a 2×2 matrix to solve this problem.

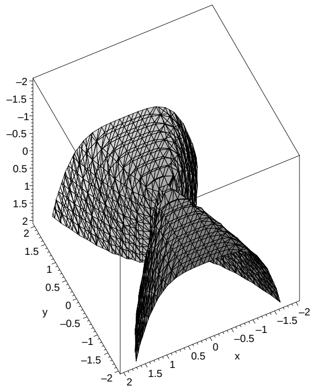

Fig. 2. Buchberger Example

In fact, the support computed by ISPOS is optimal, even without applying the implicit degree bound, since the implicit equation of this surface is given by

$$x^4 - y^2z = 0.$$

6.4 Example from [Bus01]

$$x = \frac{s^2}{s^3+t^3} \; y = \frac{s^3}{s^3+t^3} \; z = \frac{t^2}{s^3+t^3}$$

Our algorithm yields the monomials

$$[x^3y, x^3, x^3y^2, y^2z^3]$$

To finish the implicitization by the method of [CGKW01] we start with the vector $[x^3y, x^3, x^3y^2, y^2z^3]$ and construct the 4×4 matrix

$$\begin{bmatrix} x^6y^2 & x^6y & x^6y^3 & x^3y^3z^3 \\ x^6y & x^6 & x^6y^2 & x^3y^2z^3 \\ x^6y^3 & x^6y^2 & x^6y^4 & x^3y^4z^3 \\ x^3y^3z^3 & x^3y^2z^3 & x^3y^4z^3 & y^4z^6 \end{bmatrix}.$$

After substitution and integration we obtain a matrix whose nullspace is spanned by the vector $[-2, 1, 1, -1]$. This gives the implicit equation:

$$-2\,x^3y + x^3 + x^3y^2 - y^2z^3 = 0$$

Remark: In this example, we had to aid Maple in performing double integrations of the form:

$$\mathcal{I}_1 = \int\int \frac{s^a}{(s^3+t^3)^c}\,dt\,ds \quad \text{and} \quad \mathcal{I}_2 = \int\int \frac{s^a t^b}{(s^3+t^3)^c}\,dt\,ds$$

where $a, b, c \in \mathbb{N}$, by imposing a change of variables of the form $t = \tau\ s$ (which implies $dt = sd\tau$). Thus we obtain the formula

$$\mathcal{I}_1 = \int\int \frac{s^{a-3\ c-1}}{(1+\tau^3)}\,d\tau\,ds$$

and Maple is now able to find the result for $\mathcal{I}_1$. A similar formula allows for integrals of type $\mathcal{I}_2$ to be computed.

7 Conclusion & Future Work

We presented an algorithm to predict in advance the support of the implicit equation, given parametric equations of a curve, surface or hypersurface. This infromation can be subsequently used by implicitization algorithms resulting in dramatic gains in efficiency, e.g. [CGKW01].

• Bicubic Surface

The well-known bicubic surface example from CAGD (whose implicit equation is computed with the special method developed in [GV97]) representes a significant challenge for the IPSOS algorithm. The parametric equations of the bicubic surface $\mathcal{B}$ are given by:

$$\begin{array}{l} x = 3\ t\ (t-1)^2 + (s-1)^3 + 3\ s \\ y = 3\ s\ (s-1)^2 + t^3 + 3\ t \\ z = -3\ s\ (s^2-5\ s+5)\ t^3 - 3\ (s^3+6\ s^2-9\ s+1)\ t^2 + t\ (6\ s^3+9\ s^2-18\ s+3) - 3\ s\ (s-1) \end{array} \tag{5}$$

In particular, for the input points:

```
[[0, 0, 0, 0, 1], [0, 0, 0, 1, 1], [0, 0, 0, 2, 1], [0, 0, 0, 3, 1],
 [0, 0, 1, 0, 1], [0, 0, 2, 0, 1], [0, 0, 3, 0, 1], [1, 0, 0, 0, 1],
 [1, 0, 0, 1, 1], [1, 0, 0, 3, 1], [1, 0, 1, 0, 1], [1, 0, 2, 0, 1],
 [1, 0, 3, 0, 1], [0, 1, 0, 0, 1], [0, 1, 0, 1, 1], [0, 1, 0, 2, 1],
 [0, 1, 1, 0, 1], [0, 1, 1, 1, 1], [0, 1, 1, 2, 1], [0, 1, 1, 3, 1],
 [0, 1, 2, 0, 1], [0, 1, 2, 1, 1], [0, 1, 2, 2, 1], [0, 1, 2, 3, 1],
 [0, 1, 3, 1, 1], [0, 1, 3, 2, 1], [0, 1, 3, 3, 1]]
```

there are 737129 regular triangulations computed with TOPCOM. The last one is:

```
T[737129]:={{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 383M. This underlines the fact that we should not compute all of the regular triangulations but only the mixed cell subdivisions [MC00,MV99].

References

[BOT88] M. Ben-Or and P. Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. In *Proc. ACM Symp. Theory of Computing*, pages 301–309, New York, 1988. ACM Press.

[Buc88] Bruno Buchberger. Applications of Gröbner Bases in Non-Linear Computational Geometry. In J. R. Rice, editor, *Mathematical Aspects of Scientific Software*, volume 14 of *IMA Volumes in Mathematics and its applications*, pages 59–87. Springer-Verlag, 1988.

[Bus01] Laurent Busé. Residual resultant over the projective plane and the implicitization problem. In *ISSAC (London ON, Canada)*, pages 48–55. ACM, 2001.

[CE00] J.F. Canny and I.Z. Emiris. A subdivision-based algorithm for the sparse resultant. *J. ACM*, 47(3):417–451, May 2000.

[CGKW01] Robert M. Corless, Mark W. Giesbrecht, Ilias S. Kotsireas, and Stephen M. Watt. Numerical implicitization of parametric hypersurfaces with linear algebra. In *Artificial intelligence and symbolic computation (Madrid, 2000)*, pages 174–183. Springer, Berlin, 2001.

[CLO97] David Cox, John Little, and Donal O'Shea. *Ideals, varieties, and algorithms.* Springer-Verlag, New York, second edition, 1997. An introduction to computational algebraic geometry and commutative algebra.

[CLO98] David Cox, John Little, and Donal O'Shea. *Using algebraic geometry.* Springer-Verlag, New York, 1998.

[CLS99] Thomas Christof, Andreas Loebel, and M. Stoer. PORTA, version 1.3, 1999. University of Heidelberg and ZIB Berlin. http://www.zib.de/Optimization/Software/Porta.

[DE01] Carlos D'Andrea and Ioannis Z. Emiris. Computing sparse projection operators. In *Symbolic computation: solving equations in algebra, geometry, and engineering (South Hadley, MA, 2000)*, pages 121–139. Amer. Math. Soc., Providence, RI, 2001.

[DL00] Jesús A. De Loera. PUNTOS, http://www.math.ucdavis.edu/ deloera/, 2000.

[EC95] I.Z. Emiris and J.F. Canny. Efficient incremental algorithms for the sparse resultant and the mixed volume. *J. Symbolic Computation*, 20(2):117–149, 1995.

[GKZ91] 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).

[GKZ94] I. M. Gel′fand, M. M. Kapranov, and A. V. Zelevinsky. *Discriminants, resultants, and multidimensional determinants.* Birkhäuser Boston Inc., Boston, MA, 1994.

[GV97] Laureano Gonzalez-Vega. Implicitization of parametric curves and surfaces by using multidimensional Newton formulae. *J. Symbolic Comput.*, 23(2-3):137–151, 1997. Parametric algebraic curves and applications (Albuquerque, NM, 1995).

[MC92] Dinesh Manocha and John F. Canny. Algorithm for implicitizing rational parametric surfaces. *Comput. Aided Geom. Design*, 9(1):25–50, 1992.

[MC00] T. Michiels and R. Cools. Decomposing the secondary Cayley polytope. *Discrete Comput. Geom.*, 23(3):367–380, 2000.

[MV99] T. Michiels and J. Verschelde. Enumerating regular mixed-cell configurations. *Discrete Comput. Geom.*, 21(4):569–579, 1999.

[Ram02] Jörg Rambau. TOPCOM: Triangulations of point configurations and oriented matroids. Technical Report 02-17, ZIB, Berlin, 2002. To appear in *Proc. Intern. Congress Math. Software 2002.* http://www.zib.de/rambau/TOPCOM.

[Stu94] Bernd Sturmfels. On the Newton polytope of the resultant. *J. Algebraic Combin.*, 3(2):207–236, 1994.

[Wil02] Doran Wilde. PolyLib, http://icps.u-strasbg.fr/PolyLib/, 2002.

[Zip93] R. Zippel. *Effective Polynomial Computation.* Kluwer Academic Publishers, Boston, 1993.