Visualisation of implicit algebraic curves

Lionel Alberti, Bernard Mourrain

Visualisation of implicit algebraic curves

L. Alberti⋆◦ & B. Mourrain⋆

GALAAD

⋆ INRIA
BP 93
06902 Sophia Antipolis
France

◦ Lab. J. A. Dieudonné
Parc Valrose
06034 Nice cedex
France

Abstract

We describe a new algorithm for the visualisation of implicit algebraic curves, which isolates the singular points, compute the topological degree around these points in order to check that the topology of the curve can be deduced from the points on the boundary of these singular regions. The other regions are divided into x or y regular regions, in which the branches of the curve are also determined from information on the boundary. Combined with enveloping techniques of the polynomial represented in the Bernstein basis, it is shown on examples that this algorithm is able to render curves defined by high degree polynomials with large coefficients, to identify regions of interest and to zoom safely on these regions.

The problem of analysing and visualisation of implicit curve appears in many applications of Geometric Modeling. Fundamental operations, such as computing the intersection curve of two parametric surfaces lead to the analysis of implicit curves in the parameter domain. In Mathematics, many questions such as counting the number of isotopy types of algebraic curves of a given degree d remain open even in small degree ($d \geq 8$). Since the singularity structures of implicit models may coexists on different scales, it is important to develop multilevel approaches which allow to adapt the precision of the computation hierarchically.

In this paper, we describe a new algorithm to render accurately an implicit planar curve $\mathcal{C}$ defined by $f = 0$ ($f \in \mathbb{Q}[x, y]$) inside a given rectangular domain $\mathcal{D}_0 = [a, b] \times [c, d] \subset \mathbb{R}^2$. One of our objectives is to produce a piecewise linear approximation (within a given Hausdorff distance $\epsilon > 0$) of the curve $\mathcal{C}$, which is topologically equivalent (ie. homeomorphic) to $\mathcal{C}$. The algorithm should be able to detect "pixels" which contain a singular point and to zoom on them accurately. This requirement makes the physically based methods like the one described in [26] unsuitable to our ends.
The other feature we look for is the certified and efficient treatment of curves defined by polynomials with large coefficients or large degree. Such curves appear, for instance, when applying computer algebra techniques on exact representations of geometric objects.
There are two main types of algorithm that meet these two requirements.

The first type is inspired by the Cylindrical Algebraic Decomposition [5] algorithm. They use projection techniques based on a conceptual sweeping line perpendicular to some axis that detects the critical topological events, such as tangents to the sweeping planes and singularities. They involve the exact computation of critical points and genericity condition tests and adjacency tests. The approach has been applied successfully to curves in 2D, and even in 3D, 4D [13, 16, 12, 11, 2] and extended to surfaces [4, 23].

However, they assume exact input equations and rely on the analysis of the curve at the critical values of its projection. From an algebraic point of view, they involve the computation of (sub)-resultants polynomial and of their roots which are algebraic numbers. This can be a bottleneck in many examples with large degree and large coefficients, for which the resultant is difficult to compute, and its real roots even harder to manipulate.
Moreover, as these algorithms work by projection, they have to compute every point in the fibers above the points in the projection. In other words, most points that they compute are actually useless for the computation of the final topological description.
The complexity of the algorithm can also vary wildly, depending on the direction of projection we choose.

And non-degeneracy conditions have to be checked (which can be difficult by itself) to ensure the correctness of the algorithm. The problem is that the choice of projection is not at all related to the geometry of the curve.

This is why the CAD methods are hardly efficient in practice, and are facing complexity problems in higher dimension. They are also intrinsically delicate to apply using approximate computation.

The other type of methods relies on subdivision techniques of the original domain. This process is most commonly used to get approximations of the curve in terms of Hausdorff distance. The most famous family of algorithm using this approach is the marching cube algorithms family [20]. It doesn't not give any garantee on the topological correctness of its output, but it inspired some algorithms that do certified that their output has the same topology as the curve (usually in the smooth case). They have already been used for solving several complicated equations. See [29, 8] and the recent improvements proposed in [21], exploiting preconditioning techniques. Extensions of this approach to higher dimensional objects have also been considered [27, 16, 14, 28, 18, 1]. These subdivision methods usually fail when singular points exist in the domain. If a threshold on the minimal size for boxes is not set, the algorithm would run forever. Indeed at singularities, no matter the scale of approximation, the shape and topology of the algebraic objects remain similar.

Our algorithm like the one in [25] is hybrid. It combines approximation properties with certification and adaptivity. It subdivides the domain $\mathcal{D}_0$ into regular regions in which the curve is smooth and regions that may contain singular points. In the regular regions, we can approximate the curve as precisely as we want and the "singular" regions can be made as small as required. The algorithm computes the topology inside the regions by using what happens on their boundary and we use enveloping techniques to efficiently treat large input equations. This scheme is refined into two concrete algorithms, one being purely numerical and the other one using some algebraic computations.
This method combines the advantages of subdvision and CAD-like methods. Its complexity is intrinsic to the geometry of the curve (like the subdivision methods) and it avoids the main drawback of projections methods because it does not need to lift points.

1 General overview

In this section we set the notations and definitions and give an outline of how our method works. The rest of the article is devoted to describing precisely the different steps in the method.

Here are some notations that we will need hereafter. The implicit curve is defined by a squarefree polynomial $f \in \mathbb{Q}[x, y]$. We denote by $\mathcal{Z}(f) = \{(x, y) \in \mathbb{R}^2 | f(x, y) = 0\}$ or $\mathcal{C}$ the locus of zeroes of f. The domain in which we want to analyse the curve $\mathcal{C}$ is $\mathcal{D}_0 := [a, b] \times [c, d] \subset \mathbb{R}^2$.

The set of *singular points* of $\mathcal{C}$ is denoted $\mathcal{S} := \{(x, y) \in \mathbb{R}^2 | f(x, y) = \partial_x f(x, y) = \partial_y f(x, y) = 0\}$.

The set of *critical points* or *extremal points* of f is denoted $\mathcal{Z}_e(f) := \{(x, y) \in \mathbb{R}^2 | \partial_x f(x, y) = \partial_y f(x, y) = 0\}$.

For a subset $S \subset \mathbb{R}^2$, we denote by S°. its interior, by $\overline{S}$ its closure, and by ∂S its boundary. We call domain any closed set $\mathcal{D}$ such that $\overline{\mathcal{D}^\circ} = \mathcal{D}$ and $\mathcal{D}$ is simply connected.

We call branch (relative to a domain $\mathcal{D}$), any smooth closed segment (ie. C^∞ diffeomorphic to $[0, 1]$) whose endpoints are on $\partial\mathcal{D}$.

We call half branch at a point $p \in \mathcal{D}^\circ$ or half branch originating from $p \in \mathcal{D}^\circ$, any smooth closed segment which has one endpoint on $\partial\mathcal{D}$ and whose other endpoint is p.

Our objective is to determine the topology of $\mathcal{C}$ inside $\mathcal{D}_0$. To do this, we find a partition of $\mathcal{D}_0$ into what we call **simple domains** $\mathcal{D}_i$ for which we can compute the topology. For each kind of simple domains, we have a so-called *connection algorithms* that finds a piecewise linear approximation of the curve inside the simple domains of that type. To be able to reconstruct the global topology in $\mathcal{D}_0$ we have to ensure that the approximations on the $\mathcal{D}_i$ agree on the boundaries. Our connections algorithms have this property at no extra cost.
Our approach is iterative, which means we do not construct a partition in simple domains in one go. Instead we guess such a partition, test it, and if it doesn't work, we refine it by splitting the subdomains that are not yet simple domains. Each type of simple domain is defined by a set of *type conditions* and we have test algorithms to effectively check them.

We distinguish three different types of simple domains:

- A domain $\mathcal{D}$ is *x-regular* if it meets the x-regularity condition $\mathcal{Z}(\partial_y f) \cap \mathcal{D} = \emptyset$. In other words, there is no point on the curve which has a vertical tangent or is singular (see fig. 1 for an example).

- A domain $\mathcal{D}$ is *y-regular* if it meets the y-regularity condition $\mathcal{Z}(\partial_x f) \cap \mathcal{D} = \emptyset$. In other words, there is no point on the curve which has a horizontal tangent or is singular.

- A domain $\mathcal{D}$ is *simply singular* if $\mathcal{S} \cap \mathcal{D} = \{p\}$ and if the number n of half branches of $\mathcal{C}$ at the singular point p is equal to $\sharp(\partial\mathcal{D} \cap \mathcal{C})$, the number of points of $\mathcal{C}$ on the boundary of $\mathcal{D}$.

To end this section we give a description of the content of the remaining sections : Section 2 contains the description of a connection algorithm for x-regular and y-regular domains and a test for the x and y-regularity conditions based on Bernstein basis representation of polynomials. In section 3, we introduce the topological degree in order to compute the number n of half branches originating from a singular point, from this we deduce a test of regularity. Section 4 puts together the elements introduced in the previous sections and describes two different strategies to find a partition of $\mathcal{D}$ in simple domains. We isolate the roots of a bivariate polynomial system, using either a Bernstein subdivision solver to approximate efficiently $\mathcal{C}$ or algebraic techniques to certify the result. Section 5 shows some experimental results.

2 Regularity criterion

In this section, we consider a curve $\mathcal{C}$ in $\mathbb{R}^2$, defined by the equation $f(x, y) = 0$ with $f \in \mathbb{Q}[x, y]$ and a domain $\mathcal{D} = [a, b] \times [c, d] \subset \mathbb{R}^2$.

We recall that a tangent to the curve $\mathcal{C}$ is a line, which intersects $\mathcal{C}$ with multiplicity ≥ 2. In particular, any line through a singular point of $\mathcal{C}$ is tangent to $\mathcal{C}$.

Definition 2.1 *We say that the curve $\mathcal{C}$ is x-regular (resp. y-regular) in $\mathcal{D}$, if $\mathcal{C}$ has no tangent orthogonal to the horizontal x-direction (resp. vertical y-direction) in $\mathcal{D}$.*

Notice that if $\mathcal{C}$ is x-regular (or y-regular) it is smooth in $\mathcal{D}$ since it cannot have singular points in $\mathcal{D}$. A curve is *regular* in $\mathcal{D}$, if it is x-regular *or* y-regular in $\mathcal{D}$.

We are going to show that if $\mathcal{C}$ is x-regular in $\mathcal{D}$, then its topology can be deduced from its intersection with the boundary $\partial\mathcal{D}$. By symmetry the same applies when $\mathcal{C}$ is y-regular.

Definition 2.2 *For a point $p \in \mathcal{C} \cap \partial\mathcal{D}$, we define its interior tangent $T_p^i(\mathcal{C})$ as the tangent of $\mathcal{C}$ at p, pointing inside $\mathcal{D}$.*

If the curve is not tangent to $\partial\mathcal{D}$ at $p \in \partial\mathcal{D}$ and p is not a corner point of $\mathcal{D}$, this direction is defined by $\epsilon_p T_p(\mathcal{C})$ with $\epsilon_p = \mathrm{sign}(T_p(\mathcal{C}) \cdot \nu_p)$, where ν_p is the unit normal interior vector to $\partial\mathcal{D}$ at p.

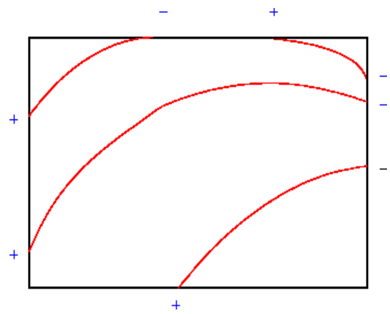

Figure 1. x**-indices of an** x**-regular domain**

If the curve is tangent we consider the half branches of $\mathcal{C} \cap \mathcal{D}$ at p, and we associate to them their unit interior tangent vector. In the following, if two opposite unit vectors are attached to a point p, we will duplicate this point, so that a point is attached to a unique interior tangent.

If p is on a corner of $\mathcal{D}$, we extend this definition as follows: We consider the cone of interior normal vectors ν_p of $\mathcal{D}$ at p, and require that $T_p^i(\mathcal{C}) \cdot \nu_p \geq 0$ for all the vectors ν_p in this cone. Thus, an interior tangent might not exist for corner points.

Definition 2.3 *For a point $p \in \mathcal{C} \cap \partial\mathcal{D}$ with interior tangent $T_p^i(\mathcal{C})$, we define its x-index (resp. y-index) as* $\mathrm{sign}(T_p^i(\mathcal{C}) \cdot e_1)$ *where e_1 is the unit vector $(1, 0) \in \mathbb{R}^2$ (resp. $(0, 1) \in \mathbb{R}^2$. If the interior tangent of $\mathcal{C}$ at p does not exist, we define the x-index of p as 0.*

For a x-regular curve $\mathcal{C}$ (with no vertical tangent) in $\mathcal{D}$ and $p \in](a, c), (a, d)[$, we have x-index$(p) = 1$. If $p \in](b, c), (b, d)[$, its x-index is -1. Moreover, if the curve is not tangent to the horizontal segment on the boundary of $\mathcal{D}$, the x-index of a point of $\mathcal{C} \cap \partial P$ which is not a corner point of $\mathcal{D}$ is not 0.

Lemma 2.4 *If $\mathcal{C}$ is x-regular in $\mathcal{D}$ (ie. $\mathcal{D}$ is a R_y domain), then a branch of $\mathcal{C} \cap \mathcal{D}$ connects a point p of x-index 1 to a point q of x-index -1, such that $x_p < x_q$.*

Proof. As the curve is x-regular, it has no vertical tangent and thus no closed loop in $\mathcal{D}$. Consequently, each of the interior connected components of $\mathcal{C} \cap \mathcal{D}$ intersects $\partial\mathcal{D}$ in two distinct points $p, q \in \mathcal{C} \cap \partial\mathcal{D}$ (with $x_p \leq x_q$).

Assume that the x-index of p, q are the same. Suppose that this index is 1. Then for an analytic parametrisation $s \in [0, 1] \mapsto (x(s), y(s))$ of the branch $[p, q]$ with $(x(0), y(0)) = p$, $(x(1), y(1)) = q$, we have $\partial_s x(0) > 0$, $\partial_s x(1) < 0$. This implies that for a value $0 < s_0 < 1$, $x(s_0) > x(1) = x_q \geq x(0) = x_p$ and that there exists $s_0' \in]0, 1[$ such that $x(s_0') = x(1)$. We deduce that $\partial_s x(s)$ vanishes in $[0, 1]$ and that the branch $[p, q]$ of $\mathcal{C}$ has a vertical tangent, which is excluded by hypothesis. If the index of p and q is -1, we exchange the role of p and q and obtain the same contradiction. As $\partial_s x(s) > 0$ for $s \in [0,1]$, we have $x_p < x_q$, which proves the lemma. □

Lemma 2.5 *Suppose that $\mathcal{C}$ is x-regular in $\mathcal{D}$ and let p, q be two consecutive points of $\mathcal{C} \cap \partial\mathcal{D}$ with*

- *q such that x_q is minimal among the points with x-index$= -1$*
- *$x_p < x_q$*

Then p, q belong to the same branch of $\mathcal{C} \cap \mathcal{D}$.

Proof. Obviously x-index$(p) = 1$ by minimality of x_q. Suppose that p, q belong to two distinct branches (p, p'), (q, q') of $\mathcal{C}$ in $\mathcal{D}$.As x-index$(q) = -1$, by the previous lemma x-index$(q') = 1$ and $x_{q'} < x_q$. And by lemma 2.4 again, x-index$(p') = -1$ and $x_p < x_{p'}$. Hence, by minimality of x_q, $x_q \le x_{p'}$. Because $x_p < x_q \le x_{p'}$, p' is between q and p on the component of $\partial\mathcal{D} - \{p, q\}$ which is to the right of p.

As the curve $\mathcal{C}$ is smooth in $\mathcal{D}$, the branches $\{p, p'\}$ and $\{q, q'\}$ do not intersect, so that q and q' are on the same connected component of $\partial\mathcal{D} - \{p, p'\}$. As $x_{q'} < x_q$, q' is between p and q on the component of $\partial\mathcal{D} - \{p, q\}$ which is to the left of q. This contradicts the fact that p and q are consecutive points of $\mathcal{C} \cap \partial\mathcal{D}$. □

Proposition 2.6 *Let $\mathcal{C} = \mathcal{Z}(f)$. If $\mathcal{D}$ is a x-regular domain, the topology of $\mathcal{C}$ in $\mathcal{D}$ is uniquely determined by its intersection $\mathcal{C} \cap \partial\mathcal{D}$ with the boundary of $\mathcal{D}$.*

Proof. We prove the proposition by induction on the number $N(\mathcal{C})$ of points on $\mathcal{C} \cap \partial_{\mathcal{D}}$, with non-zero x-index. We denote this set of points by $\mathcal{L}$.

Since the curve has no vertical tangent in $\mathcal{D}$ and has no closed loop, each of the connected components of $\mathcal{C} \cap \mathcal{D}$ intersects $\partial\mathcal{D}$ in two distinct points of x-index $\neq 0$. Thus if $N(\mathcal{C}) = 0$, then there is no branch of $\mathcal{C}$ in $\mathcal{D}$.

Assume now that $N(\mathcal{C}) > 0$, and let us show that it is possible to find two consecutive points p, q of $\mathcal{L}$ with x-index$(p) = 1$, x-index$(q) = -1$, $x_p < x_q$. As the curve $\mathcal{C}$ is smooth in $\mathcal{D}$, its k branches are not intersecting each other and they split $\mathcal{D}$ into $k+1$ connected components, which intersect the boundary $\partial\mathcal{D}$. Consider a branch $[p, q]$ which separates k of these components from the last one. Then, there is no other points of $\mathcal{C} \cap \partial\mathcal{D}$ in-between p and q. By lemma 2.5, we have x-index$(p) = 1$, x-index$(q) = -1$, $x_p < x_q$.

Removing this branch from $\mathcal{C}$, we obtain a new curve $\mathcal{C}'$ which is still x-regular and such that $N(\mathcal{C}') < N(\mathcal{C})$. We conclude by induction hypothesis, that the topology of $\mathcal{C}'$ and thus of $\mathcal{C}$ is uniquely determined. □

Proposition 2.7 *If $\mathcal{C}$ has at most one x-critical or y-critical point $\in \mathcal{D}$, which moreover is smooth, then its topology in $\mathcal{D}$ is uniquely determined by its intersection with the boundary of $\mathcal{D}$.*

Proof. Suppose that $\mathcal{C}$ has at most one x-critical point in $\mathcal{D}$, which is smooth, then the curve is smooth in $\mathcal{D}$ and has no closed loop inside $\mathcal{D}$ (otherwise the number of x-critical points would be at least 2). Therefore, the branches are intersecting $\partial\mathcal{D}$ in two points. If there is no x-critical point on a branch, by Lemma 2.4 their x-index $\in \{-1, 1\}$ are distinct. If the branch has a x-critical point, then the x-index of the end-points of the branch in $\mathcal{C}$ are the same. As the curve is smooth, the branches are not intersecting and if there are more than 2 branches there exist two p, q consecutive points of $\mathcal{C} \cap \partial\mathcal{D}$ with x-index$(p) = 1$, x-index$(q) = -1$, and $x_p < x_q$. These points belong to the same branch. Removing this branch from $\mathcal{C}$ and processing recursively in this way, we end up either with no point on $\partial\mathcal{D}$ or two points on $\partial\mathcal{D}$ with the same x-index. These points are necessarily connected by the branch containing the x-critical point of $\mathcal{C}$ in $\mathcal{D}$. This proves the proposition. □

This leads to the following algorithm:

Algorithm 2.8 CONNECTION FOR A x-REGULAR DOMAIN

INPUT: an algebraic curve $\mathcal{C}$ and a domain $\mathcal{D} = [a, b] \times [c, d] \subset \mathbb{R}^2$ such that $\mathcal{C}$ has no vertical tangent in $\mathcal{D}$.

- Isolate the points $\mathcal{C} \cap \partial\mathcal{D}$ and compute their x-index
- Order the points of $\mathcal{C} \cap \partial\mathcal{D}$ with non-zero x-indices clockwise and store them in the circular list $\mathcal{L}$.
- While $\mathcal{L}$ is not empty,
  - Take a point q such that x_q is minimal among the points in $\mathcal{L}$ with x-index$= -1$.
  - Take the point p that follows or precedes q in $\mathcal{L}$ such that $x_p < x_q$ (thus x-index$(p) = 1$).
  - add the arc $[p, q]$ to the set $\mathcal{D}$ of branches and remove p, q from $\mathcal{L}$.

OUTPUT: the set $\mathcal{D}$ of branches of $\mathcal{C}$ in $\mathcal{D}$.

Notice that a sufficient condition for the x (resp. y) regularity of f in a domain D is that the coefficients of ∂_y (resp. $\partial_x f$) in the Bernstein basis on D are all > 0 or < 0. In this case the connection algorithm can simplified even further. See [1] for more details.

3 Singular points

In this section we deal with simple singular domains. We will assume here that $\mathcal{D}$ contains a unique critical point p of f and that the curve passes through it (ie. it is a singular point of $\mathcal{C}$). We will see in section 4, how to compute such a domain.

In the following subsection we explain how using the topological degree [19] one can count the number of half branches of $\mathcal{C}$ at p and check if it is the same as the number of points in $\partial\mathcal{D} \cap \mathcal{C}$.

Finally, in the second subsection, we show that the topology in a simple singular domains (ie. satisfying the above conditions) is conic and we derive a straightforward connection algorithm from that fact.

3.1 Topological Degree

In this section, we recall the definition of the topological degree in two dimensions and how it can be computed. See [19, 30] for more details.

Let $\mathcal{D}$ be a bounded open domain of $\mathbb{R}^2$ and $F = (f_1, f_2) : \mathcal{D} \to \mathbb{R}^2$ a bivariate function which is two times continuously differentiable in $\mathcal{D}$.
A point $p \in \mathbb{R}^2$ is said to be a *regular value* of F on $\mathcal{D}$ if the roots of the equation $F(x, y) = p$ in $\mathcal{D}$ are simple roots, i.e. the determinant of the Jacobian J_F of F at these roots is nonzero).

Definition 3.1 *Let $p \in \mathbb{R}^2$ and suppose further that the roots of the equation $F(x, y) = p$, are not located on the boundary $\partial\mathcal{D}$.*

Then the topological degree of F at p relative to $\mathcal{D}$, *denoted by* $\deg[F, \mathcal{D}, p]$, *is defined by*

$$\deg[F, \mathcal{D}, p] = \sum_{\mathbf{x} \in \mathcal{D}: F(\mathbf{x}) = q} \text{sign}\det J_F(\mathbf{x}),$$

for q a regular value of F on $\mathcal{D}$ in the connected component of $\mathbb{R}^2 - F(\partial\mathcal{D})$ containing p.

It can be proved that this construction does not depend on the regular value q in the same connected component of $\mathbb{R}^2 - F(\partial\mathcal{D})$ as p [19]. If p is a regular value of F on $\mathcal{D}$, we can take $q = p$.

Remark 3.2 *The topological degree has a geometric interpretation known as the degree of the "Gauss map". It is the number of times $F(p)$ goes around $F(\mathcal{D})$ when p goes around $\mathcal{D}$ one time. And it is negative when F reverses the orientation of $\mathcal{D}$.*
The red arrows in fig. 2 picture the $F(p)$ on the boundary. This viewpoint allows to use the strong geometric intuition behind the gradient field when F is the gradient map of f.

Let us now give a more explicit formula for computing this topological degree, which involves only information on the boundary of $\mathcal{D}$.

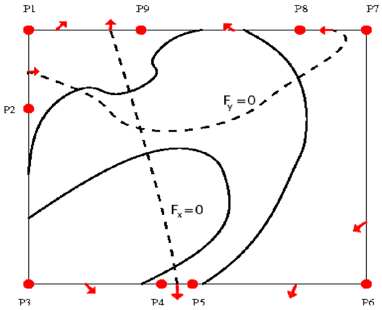

Figure 2. Computing the topological degree

Proposition 3.3 *[30] Assume here that the boundary $\mathcal{D}$ is a polygon and that it is decomposed in reverse clock-wise order into the union of segments*

$$\partial\mathcal{D} = \cup_{i=1}^{g}[p_i, p_{i+1}], \quad p_{g+1} = p_1,$$

in such a way that one of the component f_{σ_i} ($\sigma_i \in \{1, 2\}$) of $F = (f_1, f_2)$ has a constant sign ($\neq 0$) on $[p_i, p_{i+1}]$. Then

$$\deg[F, \mathcal{D}, (0, 0)] =$$

$$\frac{1}{8}\sum_{i=1}^{g}(-1)^{\sigma_i - 1}\begin{vmatrix} \text{sg}(f_{\sigma_i}(p_i)) & \text{sg}(f_{\sigma_i}(p_{i+1})) \\ \text{sg}(f_{\sigma_i+1}(p_i)) & \text{sg}(f_{\sigma_i+1}(p_{i+1})) \end{vmatrix} \quad (1)$$

where $f_1 = f_3$ and sg*(x) denotes the sign of x.*

Thus in order to compute the topological degree of F on a domain $\mathcal{D}$ bounded by a polygon, we need to separate the roots of f_1 from the roots of f_2 on $\partial\mathcal{D}$ by points $p_1, \ldots, p_{g+1}$ at which we compute the sign of f_1 and f_2. This will be performed on each segment of the boundary of $\mathcal{D}$, by a univariate root isolation method working simultaneously on f_1 and f_2, that we will described in the next section.

Figure 2 shows a sequence of points $p_1, \ldots, p_9$, which decomposes ∂D into segments on which one of the two functions ($f_1 = 0$ and $f_2 = 0$ are represented by the plain and dash curves) has a constant sign. Computing the sign of these functions and applying formula (1) yields the topological degree of $F = (f_1, f_2)$ on $\mathcal{D}$ at $(0, 0)$.

3.2 Counting the number of branches

Let us consider a curve $\mathcal{C}$ in a domain $\mathcal{D} \subset \mathbb{R}^2$, defined by the equation $f(x, y) = 0$ with $f(x, y) \in \mathbb{R}[x, y]$. Let $\nabla f = (\partial_x f, \partial_y f)$ be the gradient of f. A point $p \in \mathcal{C}$ is singular if $\nabla f(p) = 0$. We defined a real half branches of $\mathcal{C}$ at p, as a connected component of $\mathcal{C} - \{p\} \cap \mathcal{D}(p, \epsilon)$ for $\epsilon > 0$ small enough.

The topological degree of ∇f can be used to count the number of half branches at a singular point, based on the following theorem:

Theorem 3.4 (Khimshiashvili [17, 15, 31]) *Suppose that p is only root of $\nabla f = 0$ in $\mathcal{D}$. Then the number N of real half branches at p of the curve defined by $f(x,y) = f(p)$ is*

$$N = 2\,(1 - \deg[\nabla f, \mathcal{D}, (0,0)]). \qquad (2)$$

We will denote by $N(f, \mathcal{D})$ the number given by Formula (2).

In order to count the number of branches of $\mathcal{C}$ at a singular point $p \in \mathcal{C}$, first we isolate the singular point p in a domain $\mathcal{D}$, so that ∇f does not vanishes elsewhere in $\mathcal{D}$. Then we compute the topological degree $\deg[\nabla f, \mathcal{D}, (0,0)]$, as described previously, by isolating the roots of $\partial_x f$ and $\partial_y f$ on $\partial\mathcal{D}$.

Let us describe now the algorithm used to compute the topological degree of ∇f in a domain $D = [a,b] \times [c,d]$. According to formula (1), this reduces to separating the roots of $\partial_x f$ $\partial_y f$ on the boundary of D, which consists in 2 horizontal and vertical segments. The problem can thus be transformed into isolating the roots of univariate polynomials on a given interval. Hereafter, these polynomials will be called $g_1(t), g_2(t)$ and the interval $[u,v] \subset \mathbb{R}$. For instance, one the 4 cases to consider will be $g_1(t) = \partial_x f(t,c), g_2(t) = \partial_y f(t,c)$, $u = a, v = b$. We recall briefly the subdivision method described in [24, 22, 9], which can be used for this purpose. First we express our polynomials $g_1(t)$, $g_2(t)$ of degree d_1, d_2 in the Bernstein bases $(B^i_{d_k}(t;u,v))_{i=0,\ldots,d_k}$ $(k = 1,2)$, on the interval $[u,v]$:

$$g_k = \sum_{i=0}^{d_k} \lambda_{k,i}\, B^i_{d_k}(t;u,v), k = 1,2,$$

where $B^i_d(t;u,v) = \binom{d}{i}(t-u)^i(v-t)^d(v-u)^{-d}$. The number of sign variations of the sequence $\lambda_k = [\lambda_{k,0}, \ldots, \lambda_{k,d_k}]$ $(k = 1,2)$ is denoted $V(g_k;[u,v])$. By a variant of Descartes rule [3], it bounds the number of roots of g_k on the interval $[u,v]$ and is equal modulo 2 to it. Thus if $V(g_k;[u,v]) = 0$, g_k has no root in the interval $[u,v]$, if $V(g_k;[u,v]) = 1$, g_k has exactly one root in the interval $[u,v]$. This is the main ingredient of the subdivision algorithm [9], which splits the interval using de Casteljau algorithm [10] if $V(g_k;[u,v]) > 1$; store the interval if $V(g_k;[u,v]) = 1$ and remove it otherwise. It iterates the process on each subintervals until the number of sign variation is 0 or 1. The complexity analysis of the algorithm is described in [9]. See also [7].

In our case, we need to compute intervals on which one of the polynomial g_1 or g_2 has a constant sign. Thus we replace the subdivision test by the following:

- if $V(g_1;[u,v]) = 0$ or $V(g_2;[u,v]) = 0$, we store the interval $[u,v]$;
- otherwise we split it and compute the Bernstein representation of g_k $(k = 1,2)$ on the two subintervals using de Casteljau algorithm and repeat the process.

This yields the following algorithm for computing the topological degree of $\nabla f = (f_1(x,y), f_2(x,y))$ on D:

Algorithm 3.5 TOPOLOGICAL DEGREE OF (f_1, f_2)

INPUT: a polynomial $f(x,y) \in \mathbb{Q}[x,y]$ and a domain $D = [a,b] \times [c,d]$

- $\mathcal{B} := \{\}$ (a circular list representing the boundary $\partial\mathcal{D}$);
- For each side segment I of the box D,
  - Compute the restriction $g_1(t)$ (resp. $g_2(t)$) of f_1 (resp. f_2) on this side segment I and its representation in the corresponding Bernstein basis.
  - $\mathcal{L} := \{I\}$;
  - While $\mathcal{L}$ is not empty,
    - pop up an interval $[p,q]$ from $\mathcal{L}$;
    - If $V(g_1;p,q) = 0$ or $V(g_2;p,q) = 0$ insert p,q clockwise in the circular list $\mathcal{B}$;
    - otherwise split $[p,q]$ in half an insert the two subintervals in $\mathcal{L}$;
- Compute N given by formula (1) for the points in the circular list $\mathcal{B}$.

OUTPUT: N the topological degree of ∇f on D at $(0,0)$.

If we assume that $\partial_x f$ and $\partial_y f$ have no common root on the boundary of D, it can be proved (by the same arguments as those used in [3, 22, 9]) that this algorithm terminate and output a sequence of intervals on which one of the functions g_1, g_2 has no sign variation. The complexity analysis of this method is described in [24]. This analysis can be improved by exploiting the recent results in [9].

3.3 Conic structure and connection algorithm

Finally we prove that the topology in a simple singular domain $\mathcal{D}$ is conic and write a connection algorithm for theses domains.

Let $A \subset \mathbb{R}^n$ and $p \in \mathbb{R}^n$. We call cone over A with center p the set $p \star A := \bigcup_{q \in A} [p,q]$.

Proposition 3.6 *Let D be a convex simple singular domain, ie. $\mathcal{D}$ is convex such that there is a unique* *singular point s and no other critical point of f in $\mathcal{D}$, and such that the number of half branches of $\mathcal{C}$ at s is $\sharp(\partial\mathcal{D}\cap\mathcal{C})$. Then the topology of $\mathcal{D}$ is conic, ie. for any point p in the inside $\mathcal{D}$, $Z(f)\cap\mathcal{D}$ can be deformed into $p\star(\partial\mathcal{D}\cap\mathcal{C})$.*

Proof. s is the unique critical point of f in $\mathcal{D}$. If the endpoint of a half branch at s is not on $\partial\mathcal{D}$, the half branch has to be a closed loop inside $\mathcal{D}$. In that case, f would be extremal at some point p ($\neq s$) inside the loop, and p would be another critical point of f inside $\mathcal{D}$. Thus, by way of contradiction, the endpoints of half branches at s have to be on $\partial\mathcal{D}$.

The number of half branches at s is exactly $\sharp(\partial\mathcal{D}\cap\mathcal{C})$. As no two half branches can have the same endpoint on $\partial\mathcal{D}$ (that would be another singular point in $\mathcal{D}$), all points on $\partial\mathcal{D}$ are endpoints of half branches at s. Thus, at this point, we know that the connected component of s inside $\mathcal{D}$ is conic.

But in fact, there is no other connected component: Suppose we have another connected component α of $\mathcal{C}$ intersecting $\mathcal{D}$. As all points of $\partial\mathcal{D}\cap\mathcal{C}$ are connected to s, we have $\alpha\subset\mathcal{D}$. α is a smooth 1-dimensional manifold because s is the only singular point. Therefore α is a closed loop inside $\mathcal{D}$ (s might be inside it). We look at the complement of $\mathcal{C}$ in $\mathbb{R}^2$, it has a bounded connected component because one of them is inside the loop α. As f vanishes on the boundary of this component, f has an extremum inside it. This extremum cannot be s as it is in the complement of f, which is impossible. Thus, $\mathcal{C}\cap\mathcal{D}$ is connected.

This concludes our argument as we have proved that $\mathcal{C}\cap\mathcal{D}$ is equal to the connected component of s inside $\mathcal{D}$ and that it has the topology of a cone over $\partial\mathcal{D}\cap\mathcal{C}$ which is what we claimed. □

Remark 3.7 *We do not have to suppose that $\mathcal{D}$ is convex, simply connected would suffice. But we only work with convex sets (boxes) and the denomination "conic topology" originates from the convex case.*

In the end the connection algorithm is extremely simple. We just proved that the topology inside these domains is conic, that is $\mathcal{C}\cap\mathcal{D}$ can be deformed into a cone over $\mathcal{C}\cap\partial\mathcal{D}$. Therefore the connection algorithm for (convex) simply singular domains is to first compute the points q_i of $\mathcal{C}\cap\partial\mathcal{D}$, then choose an arbitrary point p inside $\mathcal{D}$ and finally for every q_i, connect q_i and p by a half branch segment $\mathfrak{b}_i=[p,q_i]$.

4 Isolating the extremal points

Let $\mathcal{D}_0=[a,b]\times[c,d]$ be a domain of $\mathbb{R}^2$. The goal of this section is to describe effective methods to partition $\mathcal{D}_0$ into simple domains. The difficult step of this approach is to isolate the roots of

$$\mathcal{Z}_e(f)=\{(x,y)\in\mathcal{D}_0,\partial_x f(x,y)=0,\partial_y(f)(x,y)=0\}.$$

which are on $\mathcal{C}$, with the following property:

- There is only one point p of $\mathcal{Z}_e(f)$ in each isolating domain $\mathcal{D}$ (and it is on $\mathcal{C}$, that is singular)
- The number of points in $\mathcal{C}\cap\partial\mathcal{D}_0$ is the number of half-branches at the singular point p (that is $N(f,\mathcal{D})=2\,(1-\deg[\nabla f,\mathcal{D},0])$).

We present two approaches. The first one exploits the Bernstein representation of f and subdivision techniques to isolate the roots of $\mathcal{Z}_e(f)$, while identifying domains where the curve is regular. It outputs an approximation of $\mathcal{C}$ to a precision that is given as input to the algorithm. We prove that, for a sufficiently high precision, the algorithm output has the same topology as $\mathcal{C}$. The second algorithm is based on algebraic techniques (namely Rational Univariate Representation) and is guaranteed to output the correct topology.

The two following methods do the isolation work in a different way but they share the test described in section 3 to count the number of half branches at a singular point.

4.1 Subdivision method

We describe here the subdivision method used to obtain such isolating domains, which a specialisation of the approach used in [21]. See also [29, 8]. This method which we recall for polynomials in $\mathbb{Q}[x,y]$ applies for general multivariate polynomials. We are going to consider the system $f(x,y)=0$, $\partial_x f(x,y)=0$, $\partial_y f(x,y)=0$ in the domain $\mathcal{D}_0=[a,b]\times[c,d]$.

Each of these polynomials is expressed in the Bernstein basis on $\mathcal{D}_0$:

$$h(x,y)=\sum_{i=0}^{d_x}\sum_{j=0}^{d_y}\gamma_{i,j}\,B^i_{d_x}(x;a,b)\,B^j_{d_y}(y;c,d),$$

where $h\in\{f,\partial_x f,\partial_y f\}$ and d_x is the degree of h in x, d_y the degree of h in y. By using a method described in [21] we can quickly generate a set of boxes where the curve is x or y-regular and small set of boxes of size smaller than a given precision $\epsilon>0$ that isolates the part of the curve where we don't yet know what is happening.
The principle of this method is to either reduce a box by using convexity inequalities on Bernstein bases or to split the boxes if the inequalities do not apply. This is the main loop of the subdivision algorithm, which is

combined with preconditioning techniques to improve the performance of the solver. The computation is iterated until the size of the box is smaller than ϵ.

When the domain is reduced in one direction, one of the functions f, $\partial_x f$, $\partial_y f$ does not vanish in the regions which are removed. Thus the curve $\mathcal{C}$ in these regions is regular and according to section 2, its topology can be deduced from the intersection of the curve with the boundary of the region.

This method can be adapted to our implicit curve problem, and yields the following algorithm:

Algorithm 4.1 Subdivision algorithm for the topology of $\mathcal{C}$

Input: a curve $\mathcal{C}$ defined by $f(x,y)=0$, $\mathcal{D}_0=[a,b]\times[c,d]$, a rendering precision $\epsilon>0$ and a computation precision ν with $\epsilon \geq \nu > 0$.

- $\mathcal{L}=\{\mathcal{D}_0\}$; $\mathcal{S}=\{\}$;
- while $\mathcal{L}\neq\emptyset$
  - Pop up a domain $\mathcal{D}$ from $\mathcal{L}$;
  - If $\mathcal{D}>\nu$, reduce or split the domain $\mathcal{D}$ according to the Bernstein coefficients of $f, \partial_x f, \partial_y f$ and insert the resulting domains in $\mathcal{L}$; apply the connection algorithm of regular domain 2.8 on the removed regions;
  - otherwise add $\mathcal{D}$ to the set of singular domains $\mathcal{S}$ and update its connected components;
- For each minimal box $\mathcal{D}$ containing a connected component of $\mathcal{S}$,
  - if $|\mathcal{D}|<\epsilon$, if $\mathcal{D}$ does not intersect such another minimal box and if $\sharp(\mathcal{C}\cap\partial\mathcal{D})=2\,(1-\deg[\nabla f,\mathcal{D},(0,0)])$, then apply the algorithm of connection 3.3 in $\mathcal{D}$;
  - otherwise replace ν by $\frac{\nu}{2}$ and apply the same algorithm on $\mathcal{D}$.

Output: A graph of points $\in\mathcal{D}$ connected by segments.

This algorithm decomposes the initial domain into regions where the topology is known and a set of non-intersecting boxes of size $\leq\epsilon$ where $\sharp(\partial\mathcal{D}\cap\mathcal{C})=2\,(1-\deg[\nabla f,\mathcal{D},0])$ (this is (2)). If ϵ corresponds to the size of a pixel, the visualisation of the curve will be correct, except in these pixel boxes, which we call singular regions. Inside them equation (2) holds, and if moreover there is a unique critical point of f, which is also on $\mathcal{C}$, then the computed topology is correct.

During the subdivision process we have to zoom on domains or equivalently to scale the variables ($x:=\lambda\,x$, $y:=\lambda\,y$). In order to handle the numerical instability problems, which may happen in this scaling step or when we have to deal polynomials with large coefficients and degrees, we use the following enveloping techniques, which allows us to compute with fixed precision numbers: To analyse the curve $\mathcal{C}$ defined by the polynomial $f\in\mathbb{Q}[x,y]$ on a domain $\mathcal{D}=I\times J$,

- we convert f to the Bernstein basis on the domain $\mathcal{D}$ using exact arithmetic:
  $f(x,y)=\sum_{i,j}\gamma_{i,j}\,B^i_{d_x}(x;I)\,B^j_{d_y}(y;J)$
- we round up and down to the nearest machine precision number $\underline{\gamma_{i,j}}\leq\gamma_{i,j}\leq\overline{\gamma_{i,j}}$, so that we have $\underline{f}(x,y)\leq f(x,y)\leq\overline{f}(x,y)$ on $\mathcal{D}$.
- We use the interval coefficients $[\underline{\gamma_{i,j}},\overline{\gamma_{i,j}}]$ to test the sign conditions and to remove the regular regions.

It can be proved that if ϵ is small enough, then this algorithm compute the topology of $\mathcal{C}$ (but for space limitation reasons, we do not include the proof here).

Remark that if $Z(f)$ is smooth in a domain D, this algorithm can be run with $\epsilon=0$ and will terminate (and output the correct topology) as every subdomain will ultimately be x-regular or y-regular.

4.2 Rational univariate representation

Choosing the precision parameter ϵ smaller than some bound was enough to certify the output of the previous algorithm. The drawback is that the bounds are difficult to compute and are bad because uniform. The algebraic technique we present hereafter, namely RUR (rational univariate representation), is guaranteed to yield the correct topology. It allows the algorithm to use coarser approximations of roots (when the critical points of f are far away from each others).

We explain in short what RUR are in the bivariate case (see [3] for more details). When given a system of equations $E=\{f_1=0,f_2=0\}$ in $\mathbb{R}^2$ with 0-dimensional solution space, it is possible to find polynomials $P,P_1,P_2\in\mathbb{R}[u]$ so that we have $\mathcal{Z}(E)=\left\{\left(\frac{P_1}{P'}(\alpha),\frac{P_2}{P'}(\alpha)\right)\mid\alpha\in\mathbb{R},\quad P(\alpha)=0\right\}$ where P is squarefree and P' is its derivative. In other words, the roots of E are the image of the roots of P by a rational map. A RUR of the roots of E can be computed by finding a separating linear function and using resultant or Groebner basis techniques.

In our case the following problem arises: $\mathcal{Z}_e(f)$ can have 1-dimensional components. Because we are dealing with curves in $\mathbb{R}^2$, we can easily separate the 1-dimensional part from the 0-dimensional part by computing $g:=\gcd(\partial_x f,\partial_y f)$. We define

$$\mathcal{Z}^1_e(f)=\mathcal{Z}(g),\quad \mathcal{Z}^0_e(f)=\mathcal{Z}\left(\frac{\partial_x f}{g},\frac{\partial_y f}{g}\right).$$

Among the points in $\mathcal{Z}^0_e(f)$ we want to be able to tell those that are in $\mathcal{C}$, that is those which are singular

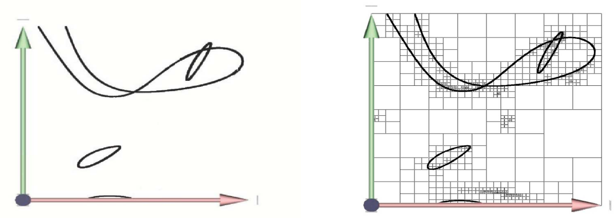

Figure 3. Self-intersection curve of a bicubic parametrized surface

points of $\mathcal{Z}(f)$. This way we can isolate the singular points of $\mathcal{C}$ from the rest of $\mathbb{N}_e(f)$. Since f is square-free, the singular locus $\mathcal{S}$ of f is 0-dimensional and $\mathcal{Z}_e^0(f) \cap \mathcal{C} = \mathcal{Z}_e^1(f) \cap \mathcal{C}$.

Therefore we compute (P, F_1, F_2) a RUR for $\mathcal{Z}_e^0(f)$ instead of $\mathcal{Z}_e(f)$ to isolate the critical points of f. And to tell which points are on $\mathcal{C}$ we compute $Q = \gcd(P, \operatorname{num} f(F_1, F_2))$ where num takes the numerator an irreducible rational fraction. It can be check easily that (Q, F_1, F_2) is a RUR for $\mathcal{Z}_e^0(f) \cap \mathcal{C}$ by using the fact that P' and P have no common roots.

Now, we use this RUR to isolate the roots of square-free polynomial P using a univariate solver (see eg. [9]). By using interval arithmetic one can find isolating intervals for the roots of $\mathcal{Z}_e^0(f)$ by computing the images of the isolating intervals of the roots of P by $F_1 := \frac{P_1}{P'}$ and $F_2 := \frac{P_2}{P'}$. This generates boxes containing these roots. If the boxes intersect we refine the isolating intervals of the roots of P until the boxes do not intersect anymore. Finally, using again interval arithmetic, we check that g does not vanish in these isolating boxes. Otherwise we refine them until it doesn't.

Keeping the boxes which correspond to roots of Q, we obtain isolating boxes which contain a single singular point. For each isolating boxes $\mathcal{D}$, we compute the topological degree. If $N(f, \mathcal{D})$ is not the number of points of $\mathcal{C} \cap \partial\mathcal{D}$, we refine the isolating box.

This yields isolating boxes for the singular points of $\mathcal{C}$, which are simply singular. The complementary of the isolating boxes is divided into boxes on which we apply the previous subdivision algorithm for smooth curves.

5 Examples

The curve in fig. 3 is the preimage in the parameter space of a self-intersection point of a bicubic surface. Its equation has been obtained by resultant computation. It is of total degree 76 and of degree 44 in each parameters. Its coefficients are of maximal bit size 590. It takes 7s to visualize this curve. Fig. 4 shows the discriminant curve of a bivariate system with few mono-

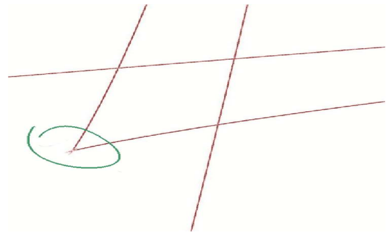

Figure 4. Curve with hidden cusp points

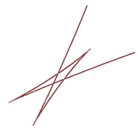

Figure 5. Hidden cusp points

mials that gives a counter-example to Kushnirenko's conjecture [6]. It is of degree 47 in x and y, and the maximal bit size of its coefficient is of order 300. It takes less that 10s to visualize it. In fig. 4 the region that has been circled looks like a cusp point, but when we blow it up in fig. 5 , we see that it is actually made of 3 cusp points and 3 crossings. The counter-example comes from this area.

Acknowledgements. This work is partially supported by AIM@SHAPE (IST Network of Excellence 506766) and ACS (IST FET Open 006413).

References

[1] L. Alberti, G. Comte, and B. Mourrain. Meshing implicit algebraic surfaces: the smooth case. In L.L. Schumaker M. Maehlen, K. Morken, editor, *Mathematical Methods for Curves and Surfaces: Tromso'04*, pages 11–26. Nashboro, 2005.

[2] J. Gerardo Alcázar and J. Rafael Sendra. Computing the topology of real algebraic space curves. *J. Symbolic Comput.*, 39:719–744, 2005.

[3] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry.* Springer-Verlag, Berlin, 2003. ISBN 3-540-00973-6.

[4] J.-S. Cheng, X.-S. Gao, and M. Li. Determining the topology of real algebraic surfaces. In *Mathematics of Surfaces*, number 3604 in LNCS, pages 121–146. Springer-Verlag, 2005.

[5] G. E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Proc. 2nd GI Conference on Automata*

Theory and Formal Languages, volume 33 of *Lecture Notes Comput. Sci.*, pages 134–183. Springer-Verlag, 1975.

[6] A. Dickenstein, M.J. Rojas, Rusekz K., Shihx, and J. Extremal real algebraic geometry and a-discriminants. Preprint, 2007.

[7] A. Eigenwillig, V. Sharma, and C. K. Yap. Almost tight recursion tree bounds for the descartes method. In *ISSAC '06: Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation*, pages 71–78. ACM Press. New York, NY, USA, 2006.

[8] G. Elber and M.-S Kim. Geometric constraint solver using multivariate rational spline functions. In *Proc. of 6th ACM Symposium on Solid Modelling and Applications*, pages 1–10. ACM Press, 2001.

[9] I. Z. Emiris, B. Mourrain, and E. P. Tsigaridas. Real algebraic numbers: Complexity analysis and experimentations. In *Reliable Implementation of Real Number Algorithms: Theory and Practice*, LNCS. Springer-Verlag, 2007. To appear (also available at).

[10] G. Farin. *Curves and surfaces for computer aided geometric design : a practical guide*. Comp. science and sci. computing. Acad. Press, 1990.

[11] G. Gatellier, A. Labrouzy, B. Mourrain, and J.-P. Técourt. Computing the topology of 3-dimensional algebraic curves. In *Computational Methods for Algebraic Spline Surfaces*, pages 27–44. Springer-Verlag, 2005.

[12] L. González-Vega and I. Necula. Efficient topology determination of implicitly defined algebraic plane curves. *Comput. Aided Geom. Design*, 19(9):719–743, 2002.

[13] T. A. Grandine and F. W. Klein. A new approach to the surface intersection problem. *Computer Aided Geometric Design*, 14:111–134, 1997.

[14] J. Hass, R. T. Farouki, C. Y. Han, X. Song, and T. W. Sederberg. Guaranteed Consistency of Surface Intersections and Trimmed Surfaces Using a Coupled Topology Resolution and Domain Decomposition Scheme. *Advances in Computational Mathematics*, 2007. To appear.

[15] T. Fukuda et W. Z. Sun K. Aoki. On the number of branches of a plane curve germ. *Kodai Math. Journal*, 9:179–187, 1986.

[16] J. Keyser, T. Culver, D. Manocha, and S. Krishnan. Efficient and exact manipulation of algebraic points and curves. *Computer-Aided Design*, 32(11):649–662, 2000.

[17] G. N. Khimšiašvili. The local degree of a smooth mapping. *Sakharth. SSR Mecn. Akad. Moambe*, 85(2):309–312, 1977.

[18] C. Liang, B. Mourrain, and J.P. Pavone. Subdivision methods for 2d and 3d implicit curves. In *Computational Methods for Algebraic Spline Surfaces*, pages 171–186. Springer-Verlag, 2007.

[19] N. G. Lloyd. *Degree theory*. Cambridge University Press, Cambridge, 1978. Cambridge Tracts in Mathematics, No. 73.

[20] W. Lorensen and H. Cline. Marching cubes: a high resolution 3d surface construction algor ithm. *Comput. Graph.*, 21(4):163–170, 1987.

[21] B. Mourrain and J.-P. Pavone. Subdivision methods for solving polynomial equations. Technical Report 5658, INRIA Sophia-Antipolis, 2005.

[22] B. Mourrain, F. Rouillier, and M.-F. Roy. *Bernstein's basis and real root isolation*, pages 459–478. Mathematical Sciences Research Institute Publications. Cambridge University Press, 2005.

[23] B. Mourrain and J.P.. Técourt. Isotopic meshing of a real algebraic surface. Technical Report 5508, INRIA Sophia-Antipolis, 2005.

[24] B. Mourrain, M. Vrahatis, and J. Yakoubsohn. On the complexity of isolating real roots and computing with certainty the topological degree. *Journal of Complexity*, 18:612–640, 2002.

[25] Helmut Ratschek and Jon G. Rokne. Scci-hybrid methods for 2d curve tracing. *Int. J. Image Graphics*, 5(3):447–480, 2005.

[26] Angela Rösch, Matthias Ruhl, and Dietmar Saupe. Interactive visualization of implicit surfaces with singularities. In *Implicit Surfaces*, pages 73–87, 1996.

[27] T. Sederberg. Algorithm for algebraic curve intersection. *Computer Aided Design*, 21:547–554, 1989.

[28] J.-K. Seong, G. Elber, and M.-S. Kim. Contouring 1- and 2-Manifolds in Arbitrary Dimensions. In *SMI'05*, pages 218–227, 2005.

[29] E. C. Sherbrooke and N. M. Patrikalakis. Computation of the solutions of nonlinear polynomial systems. *Comput. Aided Geom. Design*, 10(5):379–405, 1993.

[30] F. Stenger. Computing the topological degree of a mapping in $\mathbf{R}^n$. *Numer. Math.*, 25(1):23–38, 1975.

[31] Z. Szafraniec. On the number of branches of a 1-dimensional semianalytic set. *Kodai Math. J.*, 11(1):78–85, 1988.