Jacobi Curves: Computing the Exact Topology of Arrangements of Non-Singular Algebraic Curves *

Nicola Wolpert
Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken, Germany
nicola@mpi-sb.mpg.de

March 26, 2003

Abstract

We present an approach that extends the Bentley-Ottmann sweep-line algorithm [3] to the exact computation of the topology of arrangements induced by non-singular algebraic curves of arbitrary degrees. Algebraic curves of degree greater than 1 are difficult to handle in case one is interested in exact and efficient solutions. In general, the coordinates of intersection points of two curves are not rational but algebraic numbers and this fact has a great negative impact on the efficiency of algorithms coping with them. The most serious problem when computing arrangements of non-singular algebraic curves turns out be the detection and location of tangential intersection points of two curves. The main contribution of this paper is a solution to this problem, using only rational arithmetic. We do this by extending the concept of Jacobi curves introduced in [12]. Our algorithm is output-sensitive in the sense that the algebraic effort we need for sweeping a tangential intersection point depends on its multiplicity.

1 Introduction

Computing arrangements of curves is one of the fundamental problems in computational geometry as well as in algebraic geometry. For arrangements of lines defined by rational numbers all computations can be done over the field of rational numbers avoiding numerical errors and leading to exact mathematical results.

As soon as higher degree algebraic curves are considered, instead of linear ones, things become more difficult. In general, the intersection points of two planar curves defined by rational polynomials have irrational coordinates. That means instead of rational numbers one now has to deal with algebraic numbers. One way to overcome this difficulty is to develop algorithms that use floating point arithmetic. These algorithms are quite fast but in degenerate situations they can lead to completely wrong results because of approximation errors, rather than just slightly inaccurate outputs. Assume that for two planar curves one is interested in the number of intersection points. If the curves have tangential intersection points, the slightest inaccuracy can lead to a wrong output.

A second approach besides using floating point arithmetic is to use exact algebraic computation methods like the use of the gap theorem [5] or multivariate Sturm sequences [17]. Then of course the results are correct, but the algorithms in general are very slow.

We consider arrangements of non-singular curves in the real plane defined by rational polynomials. Although the non-singularity assumption is a strong restriction on the curves we consider, this class of curves is worthwile to be studied because of the general nature of the main problem that has to be solved. Two algebraic curves can have tangential intersec-

*Partially supported by the IST Programme of the EU as a Shared-cost RTD (FET Open) Project under Contract No IST-2000-26473 (ECG – Effective Computational Geometry for Curves and Surfaces)

tions and it is inevitable to determine them precisely in the case we are interested in exact computation. As a main tool for solving this problem we will introduce generalized Jacobi curves, for more details consider [24]. Our resulting algorithm computes the exact topology using only rational arithmetic. It is output-sensitive in the sense that the algebraic degree of the Jacobi curve that is constructed to locate a tangential intersection point depends on its multiplicity.

2 Previous work

As mentioned, methods for the calculation of arrangements of algebraic curves are an important area of research in computational geometry. For an overview consider the articles of Halperin [14] and Agarwal and Sharir [1]. A great focus is on arrangements of linear objects. Algorithms coping with linear primitives can be implemented using rational arithmetic, leading to exact mathematical results in any case. For fast filtered implementations see for example the ones in LEDA [16] and CGAL [11]. There are also some geometric methods dealing with arbitrary curves, see for example [18], [8], [22], [2], [19]. But all of them neglect the problem of exact computation in the way that they are based on an idealized real arithmetic provided by the real RAM model of computation [20]. The assumption is that all, even irrational, numbers are representable and that one can deal with them in constant time. This postulate is not in accordance with real computers.

Recently the exact computation of arrangements of non-linear objects has come into the focus of research. Wein [23] extended the CGAL implementation of planar maps to conic arcs. Berberich et al. [4] made a similar approach for conic arcs based on the improved LEDA [16] implementation of the Bentley-Ottmann sweep-line algorithm [3]. For conic arcs the problem of tangential intersection points is not serious because the coordinates of every such point are one-root expressions of rational numbers. Eigenwillig et al. [10] extended the sweep-line approach to cubic arcs. All tangential intersection points in the arrangements of cubic arcs either have coordinates that are one-root expressions or they are of multiplicity 2 and therefore can be solved using the Jacobi curve introduced in [12].

Arrangements of quadric surfaces in $\mathbb{R}^3$ are considered by Wolpert [24] and Dupont et al. [9]. By projection the first author reduces the spatial problem to the one of computing planar arrangements of algebraic curves of degree at most 4. The second authors directly work in space determining a parameterization for the intersection curve of two arbitrary implicit quadrics.

For computing planar arrangements of arbitrary planar curves very little is known. An exact approach using rational arithmetic to compute the topological configuration of a single curve is done by Sakkalis [21]. For computing arrangements of curves we are also interested in intersection points of two or more curves. Of course we could interpret these points as singular points of the curve that is the union of both. But the approach of Sakkalis for determining singular points with the help of negative polynomial remainder sequences is not very efficient, at least if singular points occur frequently.

MAPC [15] is a library for exact computation and manipulation of algebraic points. It includes a package for determining arrangements of planar curves. For degenerate situations like tangential intersections the use of the gap theorem [5] or multivariate Sturm sequences [17] is proposed. Both methods, like the one of Sakkalis, are not efficient.

3 Notation

The objects we consider and manipulate in our work are non-singular algebraic curves represented by rational polynomials. We define an *algebraic curve* in the following way: Let f be a polynomial in $\mathbb{Q}[x,y]$. We set $\text{ZERO}(f) := \{(\alpha,\beta) \in \mathbb{R}^2 \mid f(\alpha,\beta) = 0\}$ and call $\text{ZERO}(f)$ the *algebraic curve* defined by f. If the context is unambiguous, we will often identify the defining polynomial of an algebraic curve with its zero set.

For an algebraic curve f we define its *gradient vector* to be $\nabla f := (f_x, f_y) \in (\mathbb{Q}[x,y])^2$ with $f_x := \frac{\partial f}{\partial x}$. We assume the set of input curves to be *non-singular*, that means for every point $(\alpha,\beta) \in \mathbb{R}^2$ with $f(\alpha,\beta) = 0$ we have $(\nabla f)(\alpha,\beta) = (f_x(\alpha,\beta), f_y(\alpha,\beta)) \neq (0,0)$. A point (α,β) with $(\nabla f)(\alpha,\beta) = (0,0)$ we would call *singular*. The geometric interpretation is that for every point (α,β) of f there exists a unique tangent line to the curve f. This tangent line is perpendicular to $(\nabla f)(\alpha,\beta)$.

From now on we assume that all curves we consider are non-singular.

We call a point $(\alpha,\beta)\in\mathbb{R}^2$ of f *extreme* if $f_y(\alpha,\beta)=0$. Extreme points have a vertical tangent. A point $(\alpha,\beta)\in\mathbb{R}^2$ of f is named a *flex* if the curvature of f becomes zero in (α,β): $0=(f_{xx}f_y^2-2f_xf_yf_{xy}+f_{yy}f_x^2)(\alpha,\beta)$.

Two curves f and g have a *disjoint factorization* if they only share a common constant factor. Without loss of generality we assume that this is the case for every pair of curves f and g we consider during our computation. Disjoint factorization can be easily tested and established by a bivariate gcd-computation.

For two curves f and g a point (α,β) in the real plane is called an *intersection point* if it lies on f as well as on g. It is called a *tangential intersection point* of f and g if additionally the two gradient vectors are linearly dependend: $(f_xg_y-f_yg_x)(\alpha,\beta)=0$. Otherwise we speak of a *transversal intersection point*.

Last but not least we will name some properties of curves that are, unlike the previous definitions, not intrinsic to the geometry of the curves but depend on our chosen coordinate system.

We call a single curve $f=f_n(x)\cdot y^n+f_{n-1}(x)\cdot y^{n-1}+\ldots+f_0(x)\in\mathbb{Q}[x,y]$ *generally aligned* if $f_n(x)=$ constant $\neq 0$, in which case f has no vertical asymptotes. Two curves f and g are termed to be in *general relation* if every two common roots $(\alpha_1,\beta_1)\neq(\alpha_2,\beta_2)\in\mathbb{C}^2$ of f and of g have different x-values $\alpha_1\neq\alpha_2$.

We say that two pairs of curves (f_1,g_1) and (f_2,g_2) are *separate* if

1. either there are non-zero constants c_1,c_2 with $f_1=c_1\cdot f_2$ and $g_1=c_2\cdot g_2$
2. or the x-values of the complex roots of f_1 and g_1 differ pairwise from the x-values of the complex roots of f_2 and g_2.

Finally we call two curves f and g *well-behaved* if

1. f and g are both generally aligned,
2. f and g are in general relation and
3. the pairs of curves (f,g), (f,f_y), and (g,g_y) are pairwise separate.

We will shortly give an idea of what well-behavedness of two curves means. Let (α,β) be an intersection point of two curves f and g. We first consider the case $g=f_y$. If f and f_y are well-behaved, there exists a vertical stripe $a\le x\le b$ with $a<\alpha<b$ such that (α,β) is the only extreme point of f inside the stripe and the stripe contains no extreme point of f_y (and no singular point of f_y). Especially this means that flexes of f do not have a vertical tangent (Figure 1).

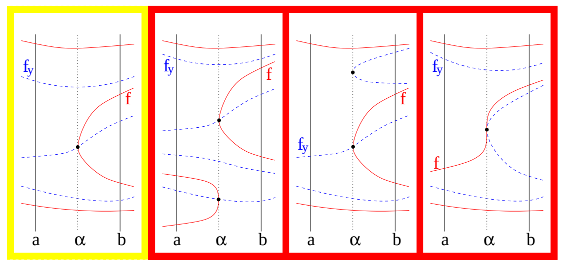

Figure 1: In the first picture the curves f and f_y are well-behaved, in the following three pictures they are not.

Next consider the case that neither f nor g is a constant multiple of the partial derivative of the other: there are no constants c_1,c_2 with $f=c_1\cdot g_y$ or $g=c\cdot f_y$. If f and g are well-behaved, then there exists a vertical stripe $a\le x\le b$ with $a<\alpha<b$ that contains exactly one intersection point of f and g, namely (α,β), and there is no extreme point of f or g inside this stripe. Especially this means that f and g do not intersect in extreme points (Figure 2).

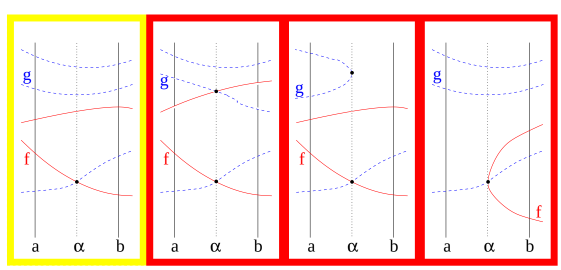

Figure 2: In the first picture the curves f and g are well-behaved, in the following three pictures they are not.

Observe that well-behavedness only depends on the given coordinate system and not on the geometry of the input curves. A random shear at the beginning will establish well-behaved input-curves with high probability. We will describe in section 8 how to test whether a pair of curves is well-behaved.

4 The overall approach

We are interested in the topology of a planar arrangement of a set F of n non-singular input curves. The curves partition the affine space in a natural way into three different types of maximal connected regions of dimensions 2, 1, and 0 called faces, edges, and vertices, respectively.

We want to compute the arrangement with a sweep-line algorithm. At each time during the sweep the branches of the curves intersect the sweep-line in some order. While moving the sweep-line along the x-axis a change in the topology of the arrangement takes place if this ordering changes. This happens if at least two different curves $f, g \in F$ intersect, or if two new branches of a curve $f \in F$ start, or if two branches of a curve $f \in F$ end. For illustration have a look at Figure 3.

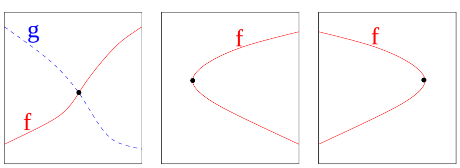

Figure 3: Two curves f and g intersect, two new branches of a curve f start, or two branches of f end.

Points of f where two branches start or end necessarily have a vertical tangent, that means they are intersection points of f and f_y. This leads to the following definition of points on the x-axis that force the sweep-line to stop and to recompute the ordering of the curves:

Definition 1 *The* event points *of a planar arrangement induced by a set F of non-singular planar curves are defined as the intersection points of each two curves $f, g \in F$ and as the intersection points of f and f_y for all $f \in F$.*

Our main algorithmic approach follows the ideas of the Bentley-Ottman sweep [3]. We hold up an $X-$ and a Y-structure. The X-structure contains the x-coordinates of event points. In the Y-structure we maintain the ordering of the curves along the sweep-line. At the beginning we found that for every $f \in F$ the curves f and f_y are well-behaved. For details on how to test this attribute and, if necessary, how to establish it with a random shear (which has no effect on the topology of the arrangement) we refer to Section 8. We insert the x-coordinates of all extreme points into the empty X-structure. We shortly remark that there can be event points left to the leftmost extreme point. This can be resolved by moving the sweep-line to the left until all pairs of adjacent curves in the Y-structure have their intersection points to the right.

If the sweep-line reaches the next event point we stop, identify the pairs of curves that intersect, the kind of intersection they have and their involved branches, recompute the ordering of the curves along the sweep-line, and according to this we update the Y-structure. If two curves become adjacent that were not adjacent in the past, we test whether they are well-behaved, see Section 8. If f and g are not well-behaved, we shear the whole arrangement and start from the beginning. Otherwise we compute the x-coordinates of their intersection points and insert them into the X-structure.

5 The X-structure

In order to make the overall approach compute the exact mathematical result in every case there are some problems that have to be solved. Describing the sweep we stated that one of the fundamental operations is the following: For two well-behaved curves f and g insert the x-coordinates of their intersection points into the X-structure. A well known algebraic method is the resultant computation of f and g with respect to y [7]. We can compute a polynomial $\mathrm{res}(f,g) \in \mathbb{Q}[x]$ of degree at most $\deg(f) \cdot \deg(g)$ with the following property:

Proposition 1 *Let $f, g \in \mathbb{Q}[x,y]$ be generally aligned curves that are in general relation. A number $\alpha \in \mathbb{R}$ is a root of $res(f,g)$ if and only if there exists exactly one $\beta \in \mathbb{C}$ such that $f(\alpha,\beta) = g(\alpha,\beta) = 0$ and $\beta \in \mathbb{R}$.*

The x-coordinates of real intersection points of f and g are exactly the real roots of the resultant polynomial $\mathrm{res}(f,g)$. Unfortunately, the intersection points of algebraic curves in general have irrational coordinates. By definition, every root of $\mathrm{res}(f,g)$ is an *algebraic number*. For $\deg(\mathrm{res}(f,g)) > 2$ there is no general way via radicals to explicitly compute the algebraic numbers in every case. But we can determine an *isolating interval* for each real root α of $\mathrm{res}(f,g)$, for example with the algorithm of Uspensky [6]. We compute two rational numbers a and b such that α is the one and only real root of $\mathrm{res}(f,g)$ in $[a,b]$. The pair $(\mathrm{res}(f,g),[a,b])$ yields a non-ambiguous rational representation of α. Of course in this representation the entry $\mathrm{res}(f,g)$ could be exchanged by any rational factor $p \in \mathbb{Q}[x]$ of $\mathrm{res}(f,g)$ with $p(\alpha) = 0$. Additionally we like α to remember the two curves f and g it originates from. We end up with inserting a representation $(p,[a,b],f,g)$ for every event point induced by f and g into the X-structure. Remark that several pairs of curves can intersect at the event point $x = \alpha$. In this case there are several representations of the algebraic number α in the X-structure, one for each pair of intersecting curves.

During the sweep we frequently have to determine the next coming event point. In order to support this query with the help of the isolating intervals we finally have to ensure the following invariant: Every two entries in the X-structure either respresent the same algebraic number, and in this case the isolating intervals in their representation are identical, or their isolating intervals are disjoint. This can be easily achieved using gcd-computation of the defining univariate polynomials and bisection by midpoints of the isolating intervals.

6 The Y-structure

A second problem that has to be solved is how to update the Y-structure at an event point. At an event point we have to stop with the sweep-line, identify the pairs of curves that intersect and their involved branches, and recompute the ordering of the curves along the sweep-line. As we have seen, the x-coordinate α of an event point is represented by at least one entry of the form $(p,[a,b],f,g)$ in the x-structure. So we can directly determine the pairs of curves that intersect at $x = \alpha$. For each pair f and g of intersecting curves we have to determine their involved branches. Furthermore we have to decide whether these two branches cross or just touch, but do not cross each other. As soon as we have these two information, updating the ordering of the curves along the sweep-line is easy.

In general, event points have irrational coordinates and therefore we cannot exactly stop the sweep-line at $x = \alpha$. The only thing we can do is stopping at the rational point a to the left of α and at the rational point b to the right of α. Using a root isolation algorithm, gcd-computation of univariate polynomials, and bisection by midpoints of the separating intervals we compute the sequence of the branches of f and g along the rational line $x = a$. We do the same along the line $x = b$. Finally, we compare these two orderings. In some cases this information is sufficient to determine the kind of event point and the involved branches of the curves inside the stripe $a \leq x \leq b$. Due to our assumption of well-behavedness we can directly compute extreme points of f (consider Figure 4):

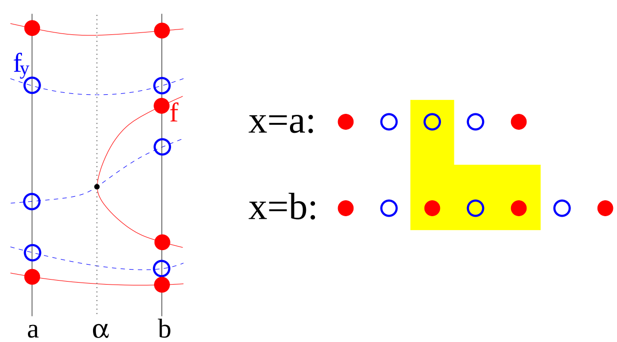

Figure 4: For computing extreme points it is sufficient to compare the sequence of f and f_y at $x = a$ to the left and at $x = b$ to the right of α.

Theorem 1 *Let $(\alpha,\beta) \in \mathbb{R}^2$ be an extreme point of a non-singular curve f and assume that f and f_y are well-behaved. We can compute two rational numbers $a \leq \alpha \leq b$ with the following property: the identification of the involved branches of f is possible by just comparing the sequence of hits of f and f_y along $x = a$ and along $x = b$.*

Proof. The x-coordinate α of the extreme point has an interval representation $(p,[a,b])$ in the X-structure. By assumption the curves f and f_y are well-behaved and therefore we know that α is not a real root of $\operatorname{res}(f_y, f_{yy}, y)$. Using bisection by midpoint we shrink the isolating interval $[a,b]$ of α until it contains no real root of $\operatorname{res}(f_y, f_{yy}, y)$.

Afterwards the vertical stripe $a \le x \le b$ contains exactly one extreme point of f, namely (α, β), and no extreme point of f_y (and no singular point of f_y). The later implies that the number and ordering of the branches of f_y does not change in the interval $[a,b]$.

The curve f has exactly one point with a vertical tangent inside the stripe and this point is not a flex. Therefore we know that the number of branches at $x = a$ differs by 2 from the one at $x = b$. A well known fact is that between every two roots of a univariate polynomial $p \in \mathbb{Q}[y]$ there is always a root of its derivative p_y. So at $x = a$ at least one branch of f_y lies between two branches of f. The same holds at $x = b$.

In order to determine the two branches of f that intersect in an extreme point we just have to compare from $-\infty$ upwards the sequences of f and f_y at $x = a$ and at $x = b$ until we detect the first difference. The branch i of f that causes this difference (either at $x = a$ or at $x = b$) intersects the $(i+1)$st branch of f in an extreme point. $\square$

Intersection points of odd multiplicity between two curves f and g are even more easy to determine (see Figure 5).

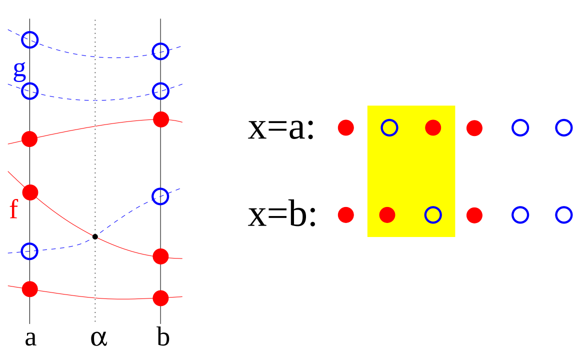

Figure 5: For computing intersection points of odd multiplicity it is sufficient to compare the sequence of f and g at $x = a$ to the left and at $x = b$ to the right of α.

Theorem 2 *Let $(\alpha, \beta) \in \mathbb{R}^2$ be an intersection point of odd multiplicity of two well-behaved non-singular curves f and g. We can compute two rational numbers $a \le \alpha \le b$ with the following property: the identification of the involved branches of f and g is possible by just comparing the sequence of hits of f and g along $x = a$ and along $x = b$.*

Proof. The x-coordinate α of the intersection point has an interval representation $(p,[a,b])$ in the X-structure. By assumption the curves f and g are well-behaved and therefore we know that α is neither a real root of $\operatorname{res}(f, f_y, y)$ nor of $\operatorname{res}(g, g_y, y)$. Using bisection by midpoint we shrink the isolating interval $[a,b]$ of α until it contains no real root of $\operatorname{res}(f, f_y, y)$ and no real root of $\operatorname{res}(g, g_y, y)$.

Afterwards the vertical stripe $a \le x \le b$ contains exactly one intersection point of f and g, namely (α, β), and no extreme point of f or g. The later implies that the ordering of the branches of f does not change in the interval $[a,b]$, nor does the ordering of the branches of g.

By assumption the intersection point of f and g has odd multiplicity and therefore we know that f and g change their ordering in (α, β).

In order to determine the branches of f and g that intersect we just have to compare from $-\infty$ upwards the sequences of f and g at $x = a$ and at $x = b$ until we detect the first difference. The branches i of f and j of g that cause this difference are the ones that intersect. $\square$

Of course this test can be easily extended to arbitrary curves under the assumption that the intersection point (α, β) is not a singular point of any of the curves.

What remains to do is locating intersection points (α, β) of even multiplicity. These points are rather difficult to locate. From the information how the curves behave slightly to the left and to the right of the intersection point we cannot draw any conclusions. At $x = a$ and at $x = b$ the branches of f and g appear in the same order, see Figure 6. We will show in the next section how to extend the idea of Jacobi curves introduced in [12] to intersection points of arbitrary multiplicity.

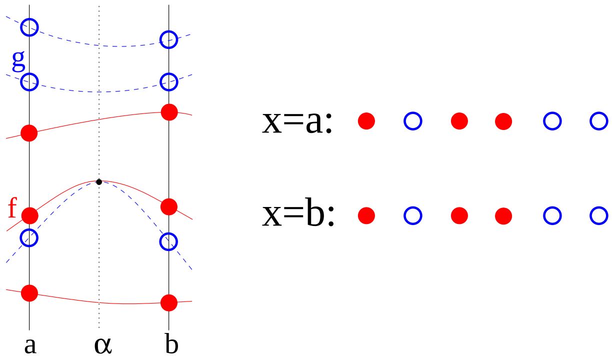

Figure 6: Intersection points of even multiplicity lead to the same sequence of f and g to the left and to the right of α.

7 The Jacobi Curves

In order to locate an intersection point of even multiplicity between two curves f and g it would be helpfull to know a third curve h that cuts f as well as g transversally in this point, see Figure 7.

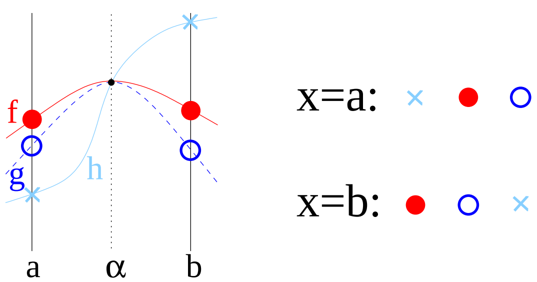

Figure 7: Introduce an auxiliary curve h in order to locate intersection points of even multiplicity of f and g.

This would reduce the problem of locating the intersection point of f and g to the easy one of locating the transversal intersection point of f and h and the transversal intersection point of g and h. In the last section we have shown how to compute the indices i, j, and k of the intersecting branches of f, g, and h, respectively. Once we have determined these indices we can conclude that the ith branch of f intersects the jth branch of g.

We will give a positive answer to the existence of transversal curves with the help of the Theorem of Implicit Functions. Let $(\alpha,\beta) \in \mathbb{R}^2$ be a real intersection point of $f, g \in \mathbb{Q}[x,y]$. We will iteratively define a sequence of polynomials $\tilde{h}_1, \tilde{h}_2, \tilde{h}_3, \ldots$ such that $\tilde{h}_k$ cuts transversally through f in (α,β) for some index k. If f and g are well-behaved, the index k is equal to the degree of α as a root of $\mathrm{res}(f,g,y)$. The result that introducing an additional curve can solve tangential intersections is already known for $k = 2$ [12]. What is new is that that this concept can be extended to every multiplicity $k > 2$. All the following results are not restricted to non-singular curves. We can determine every tangential intersection point of two arbitrary curves provided that it is not a singular point of one of the curves.

Definition 2 *Let f and g be two planar curves. We define* generalized Jacobi curves *in the following way:*

$$\begin{aligned} \tilde{h}_1 &:= g \\ \tilde{h}_{i+1} &:= (\tilde{h}_i)_x f_y - (\tilde{h}_i)_y f_x. \end{aligned}$$

Here is our main Theorem that will provide all necessary mathematical tools:

Theorem 3 *Let f and g be two algebraic curves with disjoint factorizations. Let (α,β) be an intersection point of f and g that neither is a singular point of f nor of g. There exists an index $k \geq 1$ such that $\tilde{h}_k$ cuts transversally through f in (α,β).*

Proof. In the case g cuts through f in the point (α,β), especially if (α,β) is a transversal intersection point of f and g, this is of course true for $h_1 = g$. So assume in the following that $(g_x f_y - g_y f_x)(\alpha,\beta) = \tilde{h}_2(\alpha,\beta) = 0$. From now on we will only consider the polynomials $\tilde{h}_i$ with $i \geq 2$.

By assumption we have that every point (α,β) is a non-singular point of f: $(f_x, f_y)(\alpha,\beta) \neq 0$. We only consider the case $f_y(\alpha,\beta) \neq 0$. In the case $f_x(\alpha,\beta) \neq 0$ and $f_y(\alpha,\beta) = 0$ we would proceed the same way as described in the following by just exchanging the two variables x and y. The property $f_y(\alpha,\beta) \neq 0$ leads to $(\frac{f_x}{f_y} g_y)(\alpha,\beta) = g_x(\alpha,\beta)$ and because $(g_x, g_y)(\alpha,\beta) \neq (0,0)$ we conclude $g_y(\alpha,\beta) \neq 0$. From the Theorem of Implicit Functions we derive that there are real open intervals $I_x, I_y \subset \mathbb{R}$ with $(\alpha,\beta) \in I_x \times I_y$ such that

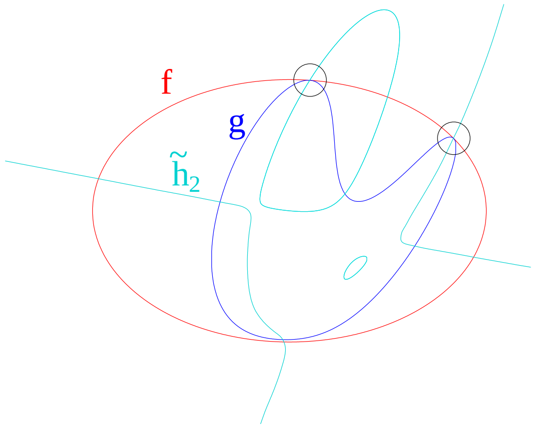

Figure 8: The second Jacobi curve $\tilde{h}_2$ cuts transversally through the intersection points of multiplicity 2 of f and g.

1. $f_y(x_0,y_0) \neq 0$ and $g_y(x_0,y_0) \neq 0$ for all $(x_0,y_0) \in I_x \times I_y$,
2. there exists a continuous function $F : I_x \to I_y$ with the two properties
   (a) $f(x,F(x)) = 0$ for all $x \in I_x$
   (b) $(x,y) \in I_x \times I_y$ with $f(x,y) = 0$ leads to $y = F(x)$,
3. and there exists a continuous function $G : I_x \to I_y$ with
   (a) $g(x,G(x)) = 0$ for all $x \in I_x$
   (b) $(x,y) \in I_x \times I_y$ with $g(x,y) = 0$ leads to $y = G(x)$.

Locally around the point (α,β) the curve defined by the polynomial f is equal to the graph of the function F. The same holds for g and G. Especially we have $\beta = F(\alpha) = G(\alpha)$. Moreover, the Theorem of Implicit Holomorphic Functions implies that F as well as G are holomorphic and thus developable in a Taylor series around the point (α,β) [13].
In the following we will sometimes consider the functions $h_i : I_x \times I_y \to \mathbb{R}$, $i \geq 2$, with

$$\begin{aligned} h_2 &:= \frac{g_x}{g_y} - \frac{f_x}{f_y} = \frac{\tilde{h}_2}{g_y f_y} \\ h_{i+1} &:= (h_i)_x - (h_i)_y \cdot \frac{f_x}{f_y} \end{aligned}$$

instead of the polynomials $\tilde{h}_i$. Each h_i is well defined for $(x,y) \in I_x \times I_y$. We have have the following relationship between the functions h_i and the polynomials $\tilde{h}_i$ defined before: For each $i \geq 2$ there exist functions $\delta_{i,2}, \delta_{i,3}, \ldots, \delta_{i,i} : I_x \times I_y \to \mathbb{R}$ such that

$$(*)\ \ h_i = \delta_{i,2} \cdot \tilde{h}_2 + \delta_{i,3} \cdot \tilde{h}_3 + \ldots + \delta_{i,i} \cdot \tilde{h}_i$$

with $\delta_{i,i}(x,y) \neq 0$ for all $(x,y) \in I_x \times I_y$. We prove this by induction on i. For $i = 2$ this is obviously true with $\delta_{2,2} = (g_y f_y)^{-1}$. The general case follows directly from the induction step:

$$\begin{aligned} h_{i+1} &= \frac{1}{f_y} \cdot ((h_i)_x f_y - (h_i)_y f_x) \\ &= \frac{1}{f_y} \cdot ((\sum_{j=2}^{i} \delta_{i,j} \tilde{h}_j)_x f_y - (\sum_{j=2}^{i} \delta_{i,j} \tilde{h}_j)_y f_x) \\ &= \frac{1}{f_y} \cdot (\sum_{j=2}^{i} \delta_{i,j} \underbrace{((\tilde{h}_j)_x f_y - (\tilde{h}_j)_y f_x)}_{=\tilde{h}_{j+1}}) \\ &\quad + \frac{1}{f_y} \cdot (\sum_{j=2}^{i} \tilde{h}_j \underbrace{((\delta_{i,j})_x f_y - (\delta_{i,j})_y f_x)}_{=:\gamma_{i,j}}) \\ &= \underbrace{\frac{1}{f_y} \gamma_{i,2}}_{:=\delta_{i+1,2}} \tilde{h}_2 + \sum_{j=3}^{i} \underbrace{\frac{1}{f_y} (\delta_{i,j-1} + \gamma_{i,j})}_{=:\delta_{i+1,j}} \tilde{h}_j \\ &\quad + \underbrace{\frac{1}{f_y} \delta_{i,i}}_{=:\delta_{i+1,i+1}} \tilde{h}_{i+1}. \end{aligned}$$

Let us assume we know the following proposition: Let $k \geq 1$. If $F^{(i)}(\alpha) = G^{(i)}(\alpha)$ for all $0 \leq i \leq k-1$, then $h_{k+1}(\alpha,\beta) = G^{(k)}(\alpha) - F^{(k)}(\alpha)$.
We know that the two polynomials f and g have disjoint factorizations. That means the Taylor series of F and G differ in some term. Remember that we consider the case that the curves defined by f and g intersect tangentially in the point (α,β). So there is an index $k \geq 2$ such that $F^{(i)}(\alpha) = G^{(i)}(\alpha)$ for all $0 \leq i \leq k-1$ and $F^{(k)}(\alpha) \neq G^{(k)}(\alpha)$. According to the proposition we have $h_{i+1}(\alpha,\beta) = G^{(i)}(\alpha) - F^{(i)}(\alpha) = 0$ for all $1 \leq i \leq k-1$. From equation (*) we inductively obtain also $\tilde{h}_{i+1}(\alpha,\beta) = 0$, $1 \leq i \leq k-1$. Especially this means that $\tilde{h}_k$ intersects f and g in (α,β). The intersection is transversal if and only if

$$((\tilde{h}_k)_x f_y - (\tilde{h}_k)_y f_x)(\alpha,\beta) = \tilde{h}_{k+1}(\alpha,\beta) \neq 0.$$

This follows easily from $0 \neq G^{(k)}(\alpha) - F^{(k)}(\alpha) = h_{k+1}(\alpha,\beta) = \delta_{k+1,k+1}(\alpha,\beta) \cdot \tilde{h}_{k+1}(\alpha,\beta)$. □

It remains to state and prove the proposition:

Proposition 2 *Let $k \geq 1$. If $F^{(i)}(\alpha) = G^{(i)}(\alpha)$ for all $0 \leq i \leq k-1$, then $h_{k+1}(\alpha,\beta) = G^{(k)}(\alpha) - F^{(k)}(\alpha)$.*

Proof. For each $i \geq 2$ we define a function $H_i : I_x \to \mathbb{R}$ by

$$H_i(x) := h_i(x, F(x)).$$

For $x = \alpha$ we derive $H_i(\alpha) = h_i(\alpha,\beta)$. So in terms of our new function we want to prove that $H_{k+1}(\alpha) = G^{(k)}(\alpha) - F^{(k)}(\alpha)$ holds if $F^{(i)}(\alpha) = G^{(i)}(\alpha)$ for all $0 \leq i \leq k-1$.
By definition we have $f(x,F(x)) : I_x \to \mathbb{R}$ and $f(x,F(x)) = 0$ for all $x \in I_x$. That means $f(x,F(x))$ is constant and therefore its derivative is equal to zero:

$$\begin{aligned}
&\frac{d}{dx} f(x,F(x)) \\
&\quad = f_x(x,F(x)) + F'(x)\, f_y(x,F(x)) \\
&\quad = 0.
\end{aligned}$$

We conclude

$$F'(x) = -\frac{f_x(x,F(x))}{f_y(x,F(x))}.$$

For the functions H_i the equality $H_i'(x) = H_{i+1}(x)$ holds, because

$$\begin{aligned}
&H_i'(x) \\
&\quad = (h_i)_x(x,F(x)) + F'(x) \cdot (h_i)_y(x,F(x)) \\
&\quad = ((h_i)_x - \frac{f_x}{f_y} \cdot (h_i)_y)(x,F(x)) \\
&\quad = h_{i+1}(x,F(x)) \\
&\quad = H_{i+1}(x).
\end{aligned}$$

Inductively we obtain $H_{k+1}(x) = H_2^{(k-1)}(x)$ for all $k \geq 1$. In order to prove the proposition it is sufficient to show the following: Let $k \geq 1$. If for all $0 \leq i \leq k-1$ we have $F^{(i)}(\alpha) = G^{(i)}(\alpha)$, then $H_2^{(k-1)}(\alpha) = (G' - F')^{(k-1)}(\alpha)$.

1. Let $k = 1$. Our assumption is $F(\alpha) = G(\alpha)$ and we have to show $H_2(\alpha) = (G' - F')(\alpha)$. We have

$$\begin{aligned}
&H_2(x) \\
&\quad = h_2(x,F(x)) \\
&\quad = \frac{g_x(x,F(x))}{g_y(x,F(x))} - \frac{f_x(x,F(x))}{f_y(x,F(x))} \\
&(G'-F')(x) \\
&\quad = \frac{g_x(x,G(x))}{g_y(x,G(x))} - \frac{f_x(x,F(x))}{f_y(x,F(x))}
\end{aligned}$$

and both functions just differ in the functions that are substituted for y in $\frac{g_x(x,y)}{g_y(x,y)}$. In the equality of $H_2(x)$ we substitute $F(x)$, whereas in the one of $(G' - F')$ we substitute $G(x)$. But of course $F(\alpha) = G(\alpha)$ leads to

$$\begin{aligned}
H_2(\alpha) &= \frac{g_x(\alpha,F(\alpha))}{g_y(\alpha,F(\alpha))} - \frac{f_x(\alpha,F(\alpha))}{f_y(\alpha,F(\alpha))} \\
&= \frac{g_x(\alpha,G(\alpha))}{g_y(\alpha,G(\alpha))} - \frac{f_x(\alpha,F(\alpha))}{f_y(\alpha,F(\alpha))} \\
&= (G'-F')(\alpha).
\end{aligned}$$

2. Let $k > 1$. We know that $F^{(i)}(\alpha) = G^{(i)}(\alpha)$ for all $0 \leq i \leq k-1$. We again use the equations

$$\begin{aligned}
H_2(x) &= \frac{g_x(x,F(x))}{g_y(x,F(x))} - \frac{f_x(x,F(x))}{f_y(x,F(x))} \\
(G'-F')(x) &= \frac{g_x(x,G(x))}{g_y(x,G(x))} - \frac{f_x(x,F(x))}{f_y(x,F(x))}
\end{aligned}$$

and the fact that $H_2(x)$ and $(G' - F')$ only differ in the functions that are substituted for y in $\frac{g_x(x,y)}{g_y(x,y)}$.
By taking $(k-1)$ times the derivative of $H_2(x)$ and $(G' - F')$, we structurally obtain the same result for both functions. The only difference is that some of the terms $F^{(i)}(x)$, $0 \leq i \leq k-1$, in H_2 are exchanged by $G^{(i)}(x)$ in $(G' - F')$. But due to our assumption we have $F^{(i)}(\alpha) = G^{(i)}(\alpha)$ for all $0 \leq i \leq k-1$ and we obtain

$$H_2^{(k-1)}(\alpha) = (G' - F')^{(k-1)}(\alpha).$$

□

We have proven that for a non-singular tangential intersection point of f and g there exists a curve $\tilde{h}_k$

that cuts both curves transversally in this point. Especially this is true for every tangential intersection point of two non-singular curves f and g. The index k depends on the degree of similarity of the functions that describe both polynomials in a small area around the given point. The degree of similarity is measured by the number of successive matching derivatives in this point. A useful result would be the following: If k is the multiplicity of α in the resultant $\mathrm{res}(f,g,y)$, then $\tilde{h}_k$ cuts transversally through f in the corresponding point (α,β). This is an immediate consequence of the previous theorem:

Corollary 1 *Let $f,g \in \mathbb{Q}[x,y]$ be two polynomials in general relation and let (α,β) be a non-singular intersection point of the curves defined by f and g. If k is the degree of α as a root of the resultant $\mathrm{res}(f,g,y)$, then $\tilde{h}_k$ cuts transversally through f.*

Proof. A well known result from algebra is that the resultant of two univariate polynomials equals the product of the differences of their roots [7]. So if we compute the resultant $X = \mathrm{res}(f,g,y)$ for two bivariate polynomials f and g, the value of X for each fixed x_0 equals the product of the differences of the roots of $f(x_0,y)$ and $g(x_0,y)$.

In the previous theorem we have proven that, in a region $I_x \times I_y$ locally around the point (α,β), we can develop f and g in Taylor series $y = F(x) = \sum_{i=0}^{\infty} \frac{F^{(i)}(\alpha)}{i!}(x-\alpha)^i$ and $y = G(x) = \sum_{i=0}^{\infty} \frac{G^{(i)}(\alpha)}{i!}(x-\alpha)^i$, respectively. The index i for which $\tilde{h}_i$ cuts transversally through f and g equals the index for which $F^{(i)}(\alpha) \neq G^{(i)}(\alpha)$ holds the first time.

For a point $x_0 \in I_x$ we know the root $F(x_0)$ of the univariate polynomials $f(x_0,y)$ and the root $G(x_0)$ of $g(x_0,y)$. So for all $x_0 \in I_x$ the term $G(x_0) - F(x_0)$ contributes to $X(x_0)$. We conclude that all roots of $G(x) - F(x)$ in I_x are roots of X, together with their multiplicities. By assumption $(\alpha, F(\alpha))$ is the only intersection point of f and g at $x = \alpha$. That means the degree k of $(x-\alpha)$ in X equals the degree of $(x-\alpha)$ in $G(x) - F(x)$. We obtain our desired result that $\tilde{h}_k$ cuts transversally through f and g. $\square$

8 Well-behaved input curves

Finally it remains to justify the well-behavedness assumption we have made for every pair of curves considered during the execution of our algorithm. We will show how to test well-behavedness. If we detect that for one pair of curves the criterion is not fulfilled, then we know that we are in a degenerate situation due to the choice of our coordinate system. In this case we stop, shear the whole set of input curves by random (for a random $v \in \mathbb{Q}$ we apply the affine transformation $\psi(x,y) = (x+vy,y)$ to each input polynomial) and restart from the beginning. A shear does not change the topology of the arrangement and we end up with pairs of well-behaved curves.

For well-behavedness of two curves f and g we have to check that

1. the pairs of curves (f,g), (f,f_y), and (g,g_y) are pairwise separate,
2. f and g are both generally aligned, and
3. that f and g are in general relation.

In order to examine the separation of two pairs of curves (f_1,g_1) and (f_2,g_2) we first look whether f_1 is a constant multiple of f_2 and g_1 is a constant multiple of g_2. This can be easily done by comparing the coefficients of the polynomials. If the answer is positive, (f_1,g_1) and (f_2,g_2) are separate. Otherwise we compute the gcd of the univariate polynomials $\mathrm{res}(f_1,g_1,y)$ and $\mathrm{res}(f_2,g_2,y)$. In the case the gcd is a constant, we know that (f_1,g_1) and (f_2,g_2) are separate. Otherwise they are not and we have to shear.

The second criterion, general alignment of f and of g, is even more easy to check by just examining the leading coefficients with respect to y.

What finally remains to do is testing whether f and g are in general relation. In case we already know that (f,g), (f,f_y), and (g,g_y) are pairwise separate and that f and g are generally aligned, this test can be easily realized with the help of the first subresultant $\mathrm{sres}_1(f,g,y) \in \mathbb{Q}[x]$ of f and g. A well-known property of this polynomial is the following [24]:

(*) A complex number α is a common root of $\mathrm{res}(f,g,y)$ and $\mathrm{sres}_1(f,g,y)$ if and only if $f(\alpha,y), g(\alpha,y) \in \mathbb{Q}[y]$ have a common factor of multiplicity ≥ 2.

Theorem 4 *Let f and g be two non-singular generally aligned curves and let (f,g), (f,f_y), and (g,g_y) be pairwise separate. Then f and g are in general relation if and only if $gcd(res(f,g,y), sres_1(f,g,y)) = constant$.*

Proof. Assume that f and g are in general relation and let α be a root of $\mathrm{res}(f,g,y)$. Then there exists exactly one $\beta \in \mathbb{C}$ with $f(\alpha,\beta) = g(\alpha,\beta) = 0$. That means for some index $i \geq 1$ the polynomial $(y-\beta)^i$ is the gcd of $f(\alpha,y)$ and $g(\alpha,y)$. We even have $i = 1$ because otherwise (α,β) would also be a root of f_y and g_y contradicting (f,g), (f,f_y), and (g,g_y) being pairwise separate. We conclude that every root α of the resultant of f and g cannot be a root of $\mathrm{sres}_1(f,g,y)$. The other direction follows easily from the property (*) of $\mathrm{sres}_1(f,g,y)$ mentioned above. □

Acknowledgements

The author would like to thank Elmar Schömer and Raimund Seidel for usefull discussions and suggestions and Arno Eigenwillig for carefully proof-reading the paper.

References

[1] P. K. Agarwal and M. Sharir. Arrangements and their applications. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 49–119. Elsevier Science Publishers B.V. North-Holland, Amsterdam, 2000.

[2] C. Bajaj and M. S. Kim. Convex hull of objects bounded by algebraic curves. *Algorithmica*, 6:533–553, 1991.

[3] J. L. Bentley and T. Ottmann. Algorithms for reporting and counting geometric intersections. *IEEE Trans. Comput.*, C-28:643–647, 1979.

[4] E. Berberich, A. Eigenwillig, M. Hemmer, S. Hert, K. Mehlhorn, and E. Schömer. A computational basis for conic arcs and boolean operations on conic polygons. In *ESA 2002, Lecture Notes in Computer Science*, pages 174–186, 2002.

[5] J. Canny. *The Complexity of Robot Motion Planning*. MIT Press, Cambridge, MA, 1987.

[6] G. E. Collins and R. Loos. Real zeros of polynomials. In B. Buchberger, G. E. Collins, and R. Loos, editors, *Computer Algebra: Symbolic and Algebraic Computation*, pages 83–94. Springer-Verlag, New York, NY, 1982.

[7] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms*. Springer, New York, 1997.

[8] D. P. Dobkin and D. L. Souvaine. Computational geometry in a curved world. *Algorithmica*, 5:421–457, 1990.

[9] L. Dupont, D. Lazard, S. Lazard, and S. Petitjean. A new algorithm for the robust intersection of two general quadrics. accepted for Symposium on Computational Geometry, 2003.

[10] A. Eigenwillig, E. Schömer, and N. Wolpert. Sweeping arrangements of cubic segments exactly and efficiently. Technical Report ECG-TR-182202-01, 2002.

[11] E. Flato, D. Halperin, I. Hanniel, and O. Nechushtan. The design and implementation of planar maps in cgal. In *Proceedings of the 3rd Workshop on Algorithm Engineering*, Lecture Notes Comput. Sci., pages 154–168, 1999.

[12] N. Geismann, M. Hemmer, and E. Schömer. Computing a 3-dimensional cell in an arrangement of quadrics: Exactly and actually! In *Proc. 17th Annu. ACM Sympos. Comput. Geom.*, pages 264–271, 2001.

[13] R. Gunning and H. Rossi. *Analytic functions of several complex variables*. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1965.

[14] D. Halperin. Arrangements. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 21, pages 389–412. CRC Press LLC, Boca Raton, FL, 1997.

[15] J. Keyser, T. Culver, D. Manocha, and S. Krishnan. MAPC: A library for efficient and exact manipulation of algebraic points and curves. In *Proc. 15th Annu. ACM Sympos. Comput. Geom.*, pages 360–369, 1999.

[16] K. Mehlhorn and S. Näher. *LEDA – A Platform for Combinatorial and Geometric Computing*. Cambridge University Press, 1999.

[17] P. S. Milne. On the solutions of a set of polynomial equations. In *Symbolic and Numerical Computation for Artificial Intelligence*, pages 89–102. 1992.

[18] K. Mulmuley. A fast planar partition algorithm, II. *J. ACM*, 38:74–103, 1991.

[19] F. Nielsen and M. Yvinec. An output-sensitive convex hull algorithm for planar objects. Technical Report 2575, Institut nationale de recherche en informatique at en automatique, INRIA Sophia-Antipolis, 1995.

[20] F. P. Preparata and M. I. Shamos. *Computational geometry and introduction*. Springer-Verlag, New York, 1985.

[21] T. Sakkalis. The topological configuration of a real algebraic curve. *Bulletin of the Australian Mathematical Society*, 43:37–50, 1991.

[22] J. Snoeyink and J. Hershberger. Sweeping arrangements of curves. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 6:309–349, 1991.

[23] R. Wein. On the planar intersection of natural quadrics. In *ESA 2002, Lecture Notes in Computer Science*, pages 884–895, 2002.

[24] N. Wolpert. *An Exact and Efficient Approach for Computing a Cell in an Arrangement of Quadrics*. Universität des Saarlandes, Saarbrücken, 2002. Ph.D. Thesis.