Nečas Center for Mathematical Modeling

Exploiting the Implicit Support Function for a Topologically Accurate Approximation of Algebraic Curves

E. Blažková, Z. Šír

Preprint no. 2014-07

http://ncmm.karlin.mff.cuni.cz/

Exploiting the Implicit Support Function for a Topologically Accurate Approximation of Algebraic Curves

Eva Blažková and Zbyněk Šír

Faculty of Mathematics and Physics, Charles University in Prague, Czech Republic
{eblazkova,sir}@karlin.mff.cuni.cz

Abstract. Describing the topology of real algebraic curves is a classical problem in computational algebraic geometry. It is usually based on algebraic techniques applied directly to the curve equation. We use the implicit support function representation for this purpose which can in certain cases considerably simplify this task. We describe possible strategies and demonstrate them on a simple example. We also exploit the implicit support function for a features-preserving approximation of the graph topologically equivalent to the curve. This contribution is meant as a first step towards an algorithm combining classical approaches with the dual description via the support function.

Key words: algebraic curve, support function, critical points, approximation, trigonometric polynomial

1 Introduction

Solution of many problems in Computer Aided Geometric Design depends on an approximation of a curve given by an implicitly defined bivariate polynomial with rational coefficients. It is very desirable to visualize the curve in any required precision, to find the number of components or to test to which component a given point belongs. All this information is fully contained in the planar graph topologically equivalent to the curve and whose vertices are points of the algebraic curve and edges correspond to regular arcs of the curve.

Known algorithms studying the topology of an algebraic curve have always two parts. First we find out the critical points and then we connect them appropriately. There are two main types of algorithms. The first type uses the same principle as the Cylindrical Algebraic Decomposition (CAD) algorithm, cf. [5, page 159]. The other approach is based on a subdivision of the given region.

Cylindrical Algebraic Decomposition based algorithms are usually divided into three phases: First find the x-coordinates of critical points of $\mathcal{C}$, then for each x_i compute the intersection points $P_{i,j}$ of $\mathcal{C}$ and the vertical line $x = x_i$ and finally for every $P_{i,j}$ determine the number of branches of $\mathcal{C}$ on the left and right and use this information to connect the points appropriately.

The main problem of these algorithms is the second phase, because the x-coordinates of the critical points are not necessarily rational numbers and therefore the polynomials $f(x_i, y)$ have non-rational coefficients. There are several methods to deal with this problem. In [12], Hong computes xy-parallel separating boxes of critical points with rational endpoints. Then he can count the branches in Phase 3 as roots of univariate polynomial with rational coefficients. In paper [7], the authors proposed a preprocessing - linear change of coordinate. The x-coordinate is transformed so that the curve is in generic position. When the curve is in generic position, the *Sturm-Habicht* sequence is used, a suitable generalization of polynomial remainder sequence, to derive the y-coordinates of critical points (Phase 2) as rational functions of their x-coordinate and also to deduce the multiplicity of the considered critical point. Another solution was given by paper [16] - they project critical points to three axes x, y and a random one. From these projections they can recover xy-parallel boxes with rational endpoints which separate the critical points. Paper [6] give the Bitstream Descartes algorithm (a variant of interval Descartes algorithm) as an efficient algorithm to isolate roots of a polynomial with non-rational coefficients. In contrast to all above algorithms, [13] replace the Sturm-Habicht sequence with a Gröbner basis and rational univariate representation, which ensure that we avoid working with polynomials with non-rational coefficients even in non-generic position.

The second type of algorithm is based on subdivision. The only certified algorithm (i.e. one which gives the correct output for every input) based on subdivision is [4]. This algorithm subdivides the region $\mathcal{D}$ into *regular regions* (the curve is smooth inside) and *regions with singular points*, which can be made sufficiently small. The topology inside the regions containing a singular point is recovered from the information on the boundary using the topological degree.

The main contribution of this paper consists in application of the (implicit) support function representation to the construction of the graph topologically equivalent to a given algebraic curve. We also consider the subsequent high precision approximation of the curve. The support function representation describes a curve as the envelope of its tangent lines, where the distance between the tangent line and the origin is specified by a function of the unit normal vector. This representation is one of the classical tools in the field of convex geometry [11]. In this representation offsetting and convolution of curves correspond to simple algebraic operations of the corresponding support functions. In addition, it provides a computationally simple way to extract curvature information [8]. Applications of this representation to problems from Computer Aided Design were foreseen in the classical paper [15] and developed in several recent publications, see e.g., [1–3, 9, 10, 14, 17, 18].

The remainder of this paper is organized as follows. Section 2 is devoted to basic definitions and results related to the (implicit) support function representation and to the topology of algebraic curves. Section 3 describes how the use of the implicit support function can contribute to the basic phases of determination of the topology of planar algebraic curves. Issues related both to the search for critical points and their connectivity are considered. In Section 4 we show how the support function representation can be exploited for an efficient approximation of segments of the curve connecting the critical points. In Section 5 we summarize our results in an algorithm and demonstrate it on a simple example.

2 Preliminaries

In this section we first recall the definitions and basic properties of the explicit and the implicit support functions. We also summarize concepts related to the determination of the topology of algebraic curves. In both cases we slightly extend standard approaches toward our goals.

2.1 Implicit support function representation of algebraic curves

For an algebraic planar curve $\mathcal{C}$ we define its support function h as a (possibly multivalued) function defined on a subset of the unit circle

$$h : \mathbb{S}^1 \supset U \to \mathbb{R}^1$$

by which is any unit normal $\mathbf{n} = (n_1, n_2)$ associated with the distance(s) from the origin to the corresponding tangent line(s) of the curve.

As proved in [18] we can recover the curve $\mathcal{C}$ from h as the envelope of the system of lines $\{\mathbf{n} \cdot \mathbf{x} - h(\mathbf{n}) = 0 : \mathbf{n} \in U\}$. This envelope is locally parameterized via the formula

$$\mathcal{C}(\mathbf{n}) = h(\mathbf{n})\mathbf{n} + \nabla_{\mathbb{S}_1} h(\mathbf{n}) = h(\mathbf{n})\mathbf{n} + \dot{h}(\mathbf{n})\mathbf{n}^{\perp} \ , \tag{1}$$

where $\nabla_{\mathbb{S}_1}$ denotes the intrinsic gradient with respect to the unit circle, which is alternatively expressed using the derivative $\dot{h}(\mathbf{n})$ with respect to the arc-length and $\mathbf{n}^{\perp}$ is the clockwise rotation of $\mathbf{n}$ about the origin by the angle $\frac{\pi}{2}$.

For an algebraic curve $\mathcal{C}$ defined as the zero set of a polynomial $f(x, y) = 0$ we typically do not obtain an explicit expression of h but rather an implicit one, which is closely related to the notion of dual curve.

Definition 1. *Let $\mathcal{C}$ be a curve in projective plane. The* dual of $\mathcal{C}$ *is the Zariski closure of the set in the dual projective plane consisting of tangent lines of $\mathcal{C}$.*

The equation of the dual curve

$$D(h, \mathbf{n}) = 0 \tag{2}$$

can be computed by eliminating x and y from the following system of equations:

$$\begin{aligned} \mathbf{n} \cdot \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)^{\perp} &= 0 \\ \mathbf{n} \cdot (x, y) &= h \ . \end{aligned} \tag{3}$$

Definition 2. *The dual equation $D(h, \mathbf{n}) = 0$ together with the algebraic constraint $n_1^2 + n_2^2 = 1$ is called the implicit definition of the support function h or simply* the implicit support function.

If the partial derivative $\partial D/\partial h$ does not vanish at $(\mathbf{n}_0, h_0)$ then (2) implicitly defines the support function

$$\mathbf{n} \mapsto h(\mathbf{n})$$

in a certain neighborhood of $(\mathbf{n}_0, h_0) \in \mathbb{R}^3$.

The (implicit) support function is obviously a kind of dual representation which takes into account the Euclidean metric. It has many nice properties. Let us recall how it is affected by selected geometric operations, cf. [15, 18]:

(i) *translation* by a translation vector $\mathbf{v} \in \mathbb{R}^2$

$$\begin{aligned} h(\mathbf{n}) &\mapsto \tilde{h}(\mathbf{n}) := h(\mathbf{n}) + \mathbf{v} \cdot \mathbf{n} \\ D(h, \mathbf{n}) = 0 &\mapsto \tilde{D}(\tilde{h}(\mathbf{n}), \mathbf{n}) := D(h(\mathbf{n}) + \mathbf{v} \cdot \mathbf{n}, \mathbf{n}) = 0 \ , \end{aligned}$$

(ii) *rotation* by an orthogonal matrix $\mathbf{A} \in SO(2)$

$$\begin{aligned} h(\mathbf{n}) &\mapsto \tilde{h}(\mathbf{n}) := h(\mathbf{A}\mathbf{n}) \\ D(h(\mathbf{n}), \mathbf{n}) = 0 &\mapsto \tilde{D}(\tilde{h}(\mathbf{n}), \mathbf{n}) := D(h(\mathbf{A}\mathbf{n}), \mathbf{n}) = 0 \ , \end{aligned}$$

(iii) *scaling* by a factor $\lambda \in \mathbb{R}$

$$\begin{aligned} h(\mathbf{n}) &\mapsto \tilde{h}(\mathbf{n}) := \lambda h(\mathbf{n}) \\ D(h(\mathbf{n}), \mathbf{n}) = 0 &\mapsto \tilde{D}(\tilde{h}(\mathbf{n}), \mathbf{n}) := D(\lambda h(\mathbf{n}), \mathbf{n}) = 0 \ , \end{aligned}$$

(iv) *offseting* with a distance $\delta \in \mathbb{R}$

$$\begin{aligned} h(\mathbf{n}) &\mapsto \tilde{h}(\mathbf{n}) := h(\mathbf{n}) + \delta \\ D(h(\mathbf{n}), \mathbf{n}) = 0 &\mapsto \tilde{D}(\tilde{h}(\mathbf{n}), \mathbf{n}) := D(h(\mathbf{n}) + \delta, \mathbf{n}) = 0 \ . \end{aligned}$$

Moreover, the support function representation is very suitable for describing the *convolution* $\mathcal{C}_3 = \mathcal{C}_1 \star \mathcal{C}_2$ of curves $\mathcal{C}_1, \mathcal{C}_2$ as this operation corresponds to the sum of the associated support functions $h_3 = h_1 + h_2$ and its implicit support function can be obtained by eliminating h_1, h_2 from the system of equations

$$D_1(h_1, \mathbf{n}) = 0, \quad D_2(h_2, \mathbf{n}) = 0 \quad \text{and} \quad h_3 = h_1 + h_2 \ ,$$

see [18, 14] for more details.

Another very useful property of the support function representation (especially in connection with G^2 Hermite interpolation problem) is that it can be efficiently used for describing the *signed curvature* of a given curve, cf. [18], in the form

$$\kappa = -\frac{1}{h + \ddot{h}} \ . \tag{4}$$

2.2 Topology of the curve

We are given a real planar algebraic curve $\mathcal{C} = \{(x, y) \in \mathbb{R}^2 \mid f(x, y) = 0\}$ where $f \in \mathbb{Q}[x, y]$. We consider the problem of determining the topology of $\mathcal{C}$. The topology of $\mathcal{C}$ is usually described by a planar graph which can have vertices at infinity and which is topologically equivalent to the original curve.

Definition 3. *Let $\mathcal{C}$ be a curve and $\mathcal{G}$ be a planar graph (possibly with vertices at infinity). The curve $\mathcal{C}$ and a graph $\mathcal{G}$ are* topologically equivalent *if and only if they are isotopic as curves of Euclidean space, i.e., there exists a continuous map $H : \mathbb{R}^2 \times [0, 1] \to \mathbb{R}^2$, such that*

- *$H(x, t)$ is a homeomorphism for all $t \in [0, 1]$,*
- *$H(x, 0) = \mathrm{id}$,*
- *$H(\mathcal{C}, 1) = \mathcal{G}$.*

Consider a vertical line l moving from the left side ($x = -\infty$) to the right ($x = \infty$). At any position there is a finite number of intersections of l and $\mathcal{C}$. The number of intersections can change only when $\mathcal{C}$ has a *critical point* on this x-coordinate. To ensure that the graph $\mathcal{G}$ is topologically equivalent to $\mathcal{C}$ we have to include all critical points among vertices of $\mathcal{G}$. Namely

Definition 4. *Let $f(x, y) \in \mathbb{Q}[x, y]$ define the real algebraic curve*

$$\mathcal{C} = \{(x, y) \in \mathbb{R}^2 \mid f(x, y) = 0\} .$$

The point $(a, b) \in \mathcal{C}$ is called

- x-critical point *if $\frac{\partial f}{\partial x} = 0$, similarly we define* y-critical point*,*
- singular point *if $\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y} = 0$,*
- x-extremal point *if $\frac{\partial f}{\partial x} = 0$ and $\frac{\partial f}{\partial y} \neq 0$, similarly we define* y-extremal point*.*

There are several methods to deal with the critical points. Our approach is related to the general scheme of Cylindrical Algebraic Decomposition (CAD) based algorithms. These algorithms are usually divided into three phases. In Phase 1 the x-coordinates of all the critical points of $\mathcal{C}$ are found. Using subresultant sequence, the discriminant $R(x)$ of f is computed. Then one determines the roots of $R(x)$ and obtain the x-coordinates (x_i, $1 \leq i \leq n$) of all critical points of $\mathcal{C}$. In Phase 2 for each x_i the intersection points $P_{i,j}$ of $\mathcal{C}$ and the vertical line $x = x_i$ are computed. These intersection points have as y-coordinates the roots of the polynomial $f(x_i, y)$. In Phase 3 the number of branches of $\mathcal{C}$ over every interval (x_i, x_{i+1}) is determined. It is the number of real roots of $f(x', y)$ for any x' from the given interval. Using this information it is possible to connect the points appropriately. In [7] a Phase 0 was proposed; a linear change of coordinate. The plane is sheared so that the curve is in generic position.

Definition 5. *The real algebraic curve $\mathcal{C}$ is in* generic position *if it satisfies the following conditions:*

- *the curve $\mathcal{C}$ has no vertical asymptotes*
- *on every vertical line $x = \alpha$, $\alpha \in \mathbb{R}$ is at most one critical point*

Obviously there are at most $\binom{c}{2}$ non-generic configurations, where c is a number of critical points. Therefore the change of coordinates is always possible.

3 The topology of the curve using the implicit support function

In this section we will discuss how the use of the implicit support function can contribute to the basic phases of determination of the topology of planar algebraic curves, see Section 2.2. We will handle certain issues related both to the search for critical points and to their connectivity.

3.1 Critical points

When the critical points are determined we can profit from the use of the support function. We devote a paragraph to every type of critical points. As we will see the support function is particularly useful in the search for cusps, points with horizontal and vertical tangents and inflections. It can also provide interesting additional information allowing us to omit self-intersections from the list of critical points while preserving the accurate curve topology. On the other hand the determination of boundary points (for a curve studied within a box) is easier on the primary curve and therefore we omit them here. An efficient global strategy would therefore be based on a combination of the information about the primary curve and its support function.

Cusps
From the general theory of algebraic curves (see e.g., [19]) the cusps on $\mathcal{C}$ correspond to inflection points in the dual representation. Cusps are distinguished as points having infinite curvature. They can be quite easily determined from the support function due to (4). If only the implicit support function is available, a condition for cusps can be formulated as follows.

Proposition 1. *Let $D(h, \mathbf{n}) = 0$ be the implicit support function of the curve $\mathcal{C}$. Then the cusps of $\mathcal{C}$ satisfy the following condition:*

$$\begin{aligned} hD_h^3 &- n_1^2(D_h^2 D_{n_2 n_2} + D_{hh} D_{n_2}^2 - 2D_h D_{hn_2} D_{n_2}) - n_1 D_h^2 D_{n_1} + \\ &+ n_2^2(D_h^2 D_{n_1 n_1} + D_{hh} D_{n_1}^2 - 2D_h D_{hn_1} D_{n_1}) - n_2 D_h^2 D_{n_2} + \\ &+ 2n_1 n_2(D_h D_{hn_2} D_{n_1} + D_h D_{hn_1} D_{n_2} + D_{hh} D_{n_1} D_{n_2} - D_h^2 D_{n_1 n_2}) = 0 \;, \end{aligned} \tag{5}$$

where the subscripts denote corresponding partial derivatives.

Proof. Using (4) we get the necessary condition for cusps

$$h(\mathbf{n}) + \ddot{h}(\mathbf{n}) = 0 \ . \tag{6}$$

Let $\mathbf{n}(s) = (n_1(s), n_2(s))$ be a parametrization of the unit circle by arc-length s and suppose that we locally have $h(\mathbf{n}(s))$. Using the chain rule we get following derivatives:

$$\dot{h} = h_{n_1}\dot{n}_1 + h_{n_2}\dot{n}_2 = -h_{n_1}n_2 + h_{n_2}n_1 \tag{7}$$

$$\begin{aligned}\ddot{h} &= h_{n_1n_1}\dot{n}_1^2 + h_{n_1n_2}\dot{n}_1\dot{n}_2 + h_{n_1}\ddot{n}_1 + h_{n_2n_2}\dot{n}_2^2 + h_{n_2n_1}\dot{n}_1\dot{n}_2 + h_{n_2}\ddot{n}_2 = \\ &= h_{n_1n_1}n_2^2 - h_{n_1n_2}n_2n_1 - h_{n_1}n_1 + h_{n_2n_2}n_1^2 - h_{n_2n_1}n_1n_2 - h_{n_2}n_2 \ ,\end{aligned} \tag{8}$$

where the dot denotes the derivative with respect to arc length s and the subscript denotes the partial derivative. The second equality in (7) and in (8) is deduced using the equality $(\dot{n}_1, \dot{n}_2) = (-n_2, n_1)$.

The partial derivatives of h can be deduced from its implicit definition. For example:

$$\frac{\partial}{\partial n_1}D(h(\mathbf{n}), n_1, n_2) = D_{n_1}(h, n_1, n_2) + h_{n_1}D_h(h, n_1, n_2) = 0 \ .$$

And therefore

$$h_{n_1} = -\frac{D_{n_1}(h, n_1, n_2)}{D_h(h, n_1, n_2)} \ .$$

Similarly we can deduce all partial derivatives of h and substitute them into (8). That equation we substitue into (6) to get a necessary condition (**??**) for cusps in variable $\mathbf{n}$. □

In concrete computations the cusps will be found by simultaneously solving equation (6) and the fundamental equations (2) and $n_1^2 + n_2^2 - 1 = 0$. The primary points are fully defined by (1).

Extremal points

Due to the dual nature of the (implicit) support function representation it is particularly easy to find the extremal points, as shown in the following

Lemma 1. *The x-extremal and y-extremal points have unit normal vectors $(\pm 1, 0)$ and $(0, \pm 1)$, respectively.*

Proof. From the definition it follows that $\frac{\partial f}{\partial x} = 0$ resp. $\frac{\partial f}{\partial y} = 0$. □

Corollary 1. *Let h be the support function implicitly defined by $D(h, \mathbf{n}) = 0$. The x and y-extremal points are the solutions of the polynomial equations in h*

$$D(h, (1, 0)) = 0 \quad \textit{and} \quad D(h, (0, 1)) = 0 \ , \tag{9}$$

respectively.

Using the envelope formula (1) we can recover extremal points on the primary curve $\mathcal{C}$.

Inflection points

Many algorithms for topologically exact description of algebraic curves do not consider inflection points. In the context of dual representations they however occur as natural splitting points. Indeed they simplify both the topology determination and subsequent approximation of individual segments.

Inflections are points where the normal vector changes its direction of movement as the point traverses the curve. Althought these can be found from the primary equation of the curve, this property is easily identified in the support function representation. Such points are of two types: the cusps and the t-extremal points of the support function, where t is the parameter on the unit circle. The first type corresponds to real inflection points, the second is the case of points at infinity. This leads to the following proposition:

Proposition 2. *Let $\mathcal{C}$ be an algebraic curve, let $t \mapsto \mathbf{n}(t)$ be a parametrization of the unit circle and consider the form $D(h,t) = 0$ of the implicit support function of $\mathcal{C}$. Then the inflection points of curve $\mathcal{C}$ are the t-critical points of the implicit support function which are neither isolated points nor self-intersections.*

We can identify the inflection points by counting the number of points of the curve on a line a little to the left and on a line a little to the right of the critical point.

Proposition 3. *Let $P = (x_0, y_0)$ be a point of the curve $\mathcal{C}$, $x_1, x_2 \in \mathbb{Q}$ and $I = [x_1, x_2]$ be an isolating interval of x_0, i.e., I does not contain other x-coordinate of x-critical point than x_0. The x-critical point P is an inflection point if and only if*

$$\sharp\{\alpha \in \mathbb{R} \mid f(x_1,\alpha) = 0\} \neq \sharp\{\alpha \in \mathbb{R} \mid f(x_0,\alpha) = 0\} \text{ or}$$
$$\sharp\{\alpha \in \mathbb{R} \mid f(x_0,\alpha) = 0\} \neq \sharp\{\alpha \in \mathbb{R} \mid f(x_2,\alpha) = 0\} ,$$

where $\sharp$ denotes the number of zeros counted with multiplicities.

Proof. We want to exclude self-intersections and isolated points, which are characterized by

$$\sharp\{\alpha \in \mathbb{R} \mid f(x_1,\alpha) = 0\} = \sharp\{\alpha \in \mathbb{R} \mid f(x_0,\alpha) = 0\} = \sharp\{\alpha \in \mathbb{R} \mid f(x_2,\alpha) = 0\} .$$

Self-intersections

Self-intersections are important features in standard algorithms for determination of the curve topology. The support function based approach however allows us to avoid the precise determination of self-intersections. From the dual point of view the two branches of the intersection are handled separately, but we can easily obtain geometrical bounds on the curve branches which certify existence and uniqueness of their intersections.

Definition 6. The tangent triangle $T(P_1, P_2)$ *is the triangle bounded by tangents at points P_1 and P_2 and by the segment P_1P_2.*

Proposition 4. *Let $\mathcal{C}_k$ be a segment of the algebraic curve $\mathcal{C}$ connecting P_1, P_2 free of cusps, inflections and extremal points. Then $\mathcal{C}_k$ lies in the interior of the tangent triangle $T(P_1, P_2)$.*

Proof. Denote by t_1 and t_2 the tangent vectors at P_1 and P_2 respectively. Due to the fact that $\mathcal{C}$ is split at extremal points and cusps, the angle between t_1 and t_2 is at most $\frac{\pi}{2}$. Therefore the arc does not intersect itself and moreover the arc does not contain any cusp, because the curve is divided in cusps. Therefore the arc is smooth and from the implicit function theorem we can suppose that the explicit formula for given arc is $c(t)$. The vector $c''(t)$ can change its sign only at cusps and inflection points and therefore it has a constant sign on the arc. Without loss of generality we can suppose that it is positive, i.e., the arc is strictly convex. From the definition of convexity, the arc lies above both tangents and below the segment P_1P_2. □

Due to the previous proposition we can find the self-intersections of the curve as the non-empty intersections of envelope triangles of all arcs in which the curve is divided. This method give us the information about which pairs of arcs intersect and also the approximate positions of the self-intersections in the intersections of envelope triangles.

Proposition 5. *Let C_1 and C_2 be two simple curve segments. If their bounding triangles $T_1 = T(P_1, P_1')$ and $T_2 = T(P_2, P_2')$ intersect in the following way:*

- *The edge P_1P_1' intersects the edge P_2P_2',*
- *$P_1, P_1' \notin T_2$ and*
- *$P_2, P_2' \notin T_1$,*

then the segments have precisely one intersection and it lies in $T_1 \cap T_2$.

Proof. Existence of the intersection follows from the transversal intersection of the triangles. The uniqueness is ensured by the convexity of both curve segments within the bounding tangent triangles. □

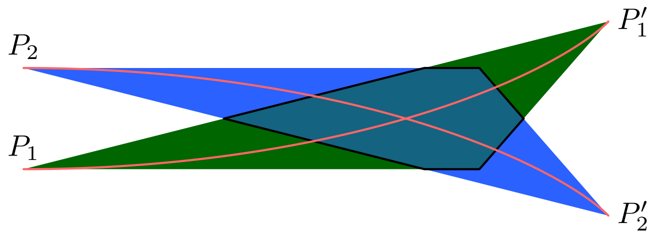

Fig. 1. Two simple curve segments and their tangent triangles. The intersection of segments lies inside the intersection of tangent triangles.

3.2 Connectivity of critical points

When we have determined the position of the critical points (Phase 1 of a general CAD based algorithm) we need to connect them appropriately. First we will study the general situation, when only the implicit formula of the curve is given. Then we describe the advantages of this approach when the given curve is an offset curve of a parametric curve.

Connectivity based on implicit support function
When we have the implicit support function of the curve, using the implicit function theorem we have also G^2 data at every point and we can profit from them. We describe some rules which the connected points have to satisfy:

1. *The difference of angles of tangents (normals) of two connected points is at most $\frac{\pi}{2}$.*
   This is because the curve is split at extremal points, cusps and inflection points.
2. *The sign of the second derivative at given point P determines in which halfplane given by the tangent line at P are the points connected to P. If the sign is negative, the points connected to P are in the same halfplane as the normal vector to $\mathcal{C}$ at P, if the sign is positive, they are in the other halfplane.*
   This rule follows immediately from the definition of convexity.

In many cases these two rules yield the connectivity of the given curve. If not, it seems that often we are able to determine the topology by subdividing (possibly several times) the maximal angle in rule 1, i.e, we add extra splitting points. For example, in the first iteration we add points with normal vector $\left(\pm\frac{1}{\sqrt{2}}, \pm\frac{1}{\sqrt{2}}\right)$. These points we can determine similarly to extremal points, see Section 3.1.

Additional connectivity information for offsets
If this general approach turns out to be insufficient we can either use one of CAD based algorithms cited in Section 1 or exploit some additional properties of studied curves. Here we would like to emphasize that in the case when the curve under examination is an offset to a given parametric curve, the connectivity is given by the parametrization. We can proceed in following steps:

1. Determine critical points on the offset curve.
2. Find the corresponding points on the original curve.
3. Connect points on the original curve by decreasing parameter.
4. Apply the same connectivity to the offset curve.

In this way the topology of the parametric curve is transferred to the offset curve.

4 Implicit support function based approximation

The support function representation can be exploited for an efficient approximation of segments of the curve connecting the critical or inflection points. Compared to approximation in the primary space it can bring several advantages, which will be discussed in this section.

Because we want to preserve features of the implicitly defined offsets and convolutions, it is suitable to interpolate the critical points up to the second order geometric data. Indeed, e.g. the cusps are distinguished by having infinite curvature. Using the support function representation it is possible to perform the G^2 Hermite interpolation by solving a system of linear equations [3]. The interpolation of critical points can be combined with an optimization of the approximation of the connection segments.

4.1 Approximation space

A suitable space of implicit support functions must be fixed in order to perform an efficient approximation.

Definition 7. *A set $\mathcal{A}$ of functions $h : \mathbb{S}^1 \to \mathbb{R}$ is called* a rational approximation space *if the following conditions hold:*

- *$\mathcal{A}$ is a real linear space of finite dimension.*
- *$\mathcal{A}$ is (as a set) invariant with respect to the rotations of $\mathbb{S}^1$.*
- *The curves with support functions from $\mathcal{A}$ are rational.*

Any segment of the primary algebraic curve will be approximated by a piece of a parametric curve with support function $h \in \mathcal{A}$. If $\{a_i\}_{i=1}^n$ is a basis of $\mathcal{A}$ then

$$h(t) = \sum_{i=1}^{n} c_i a_i(t) \ ,$$

where c_i are free coefficients. The parametric segment $\mathbf{x}_i(t)$ is computed from h via the envelope formula (1). Let us stress the fact, that in the definition of approximation space we require that the resulting segments are rational. Their union, which approximate the whole algebraic curve can therefore be represented in the NURBS format.

It was shown in [18] that suitable subspaces of trigonometric polynomials satisfy the three required conditions. In order to obtain a sufficient number of degrees of freedom for G^2 Hermite interpolation we will from now on use the trigonometric polynomials of degree 3:

$$\mathcal{A} = \mathrm{Span}\{1, \sin t, \cos t, \sin 2t, \cos 2t, \sin 3t, \cos 3t\} \ . \tag{10}$$

The main drawback of trigonometric polynomials is that they can not produce curves with inflections (and interpolate zero curvature). For an accurate (G^2) interpolation of inflections we plan to use other approximation spaces including square roots of trigonometric polynomials and more generally implicitly defined multivalued support functions. Alternatively it is possible to approximate inflections only with G^1 precision.

4.2 G^2 Hermite interpolation and fixing degrees of freedom

G^2 Hermite interpolation with trigonometric polynomials is described in detail in [3]. We will extend this procedure to points with infinite curvature (cusps) and we will also discuss how to optimize the possible free degrees of freedom.

G^2 Hermite interpolation can efficiently be performed on the level of support function due to following

Proposition 6. *Let $\mathcal{C}$ be a planar curve with support function h, defined at least locally in a neighborhood of $\mathbf{n}_0$. If g is a function defined also in a neighborhood of $\mathbf{n}_0$ and satisfying*

$$g(\mathbf{n}_0) = h(\mathbf{n}_0), \qquad \dot{g}(\mathbf{n}_0) = \dot{h}(\mathbf{n}_0), \qquad \ddot{g}(\mathbf{n}_0) = \ddot{h}(\mathbf{n}_0)\ . \tag{11}$$

Then the corresponding curve $\mathbf{x}_g$ obtained via (1) *interpolates the position of the point $\mathcal{C}(n_0)$, its normal and its curvature.*

Proof. Due to (1)

$$\mathcal{C}(\mathbf{n}_0) = h(\mathbf{n}_0)\mathbf{n}_0 + \dot{h}(\mathbf{n}_0)\mathbf{n}_0^{\perp} = g(\mathbf{n}_0)\mathbf{n}_0 + \dot{g}(\mathbf{n}_0)\mathbf{n}_0^{\perp} = \mathbf{x}_g(\mathbf{n}_0)\ .$$

The two curves have also the common normal $\mathbf{n}_0$ at their common point. Finally they have also the same curvature

$$\kappa = -\frac{1}{h(\mathbf{n}_0) + \ddot{h}(\mathbf{n}_0)} = -\frac{1}{g(\mathbf{n}_0) + \ddot{g}(\mathbf{n}_0)}$$

due to (4). □

A corollary of the previous proposition is that the G^2 Hermite interpolation in the curve space is thus reduced to the C^2 interpolation in the approximation space. The right hand sides of (11) will be obtained from $D(h, \mathbf{n})$ via implicit differentiation. Interpolation at any point thus imposes three linear conditions on coefficients c_i. More precisely, for $g(t) = \sum_{i=1}^{7} c_i a_i(t)$, an element of the approximation space (10), the conditions (11) has the following form

$$\sum_{i=1}^{7} c_i a_i(t) = h(\mathbf{n}_0), \quad \sum_{i=1}^{7} c_i a_i'(t) = \dot{h}(\mathbf{n}_0), \quad \sum_{i=1}^{7} c_i a_i''(t) = \ddot{h}(\mathbf{n}_0)\ , \tag{12}$$

where $t = \arctan\left(\frac{n_{01}}{n_{02}}\right)$. Matching the support function up to the second derivative also reproduces the cusps, which correspond to the case $h(\mathbf{n}_0) + \ddot{h}(\mathbf{n}_0) = 0$. This case, which is singular in the primary curve space, is completely regular from the point of view of the support function.

The interpolation of cusps and inflections is very important both for obtaining a low approximation error and for estimating the approximation error. In this case the error evaluates simply as the maximal error of the support function on the given interval.

Proposition 7. *Let h, g be two support functions defined on the interval $U = [\mathbf{n}_0, \mathbf{n}_1]$, such that*

$$g(\mathbf{n}_i) = h(\mathbf{n}_i), \qquad \dot{g}(\mathbf{n}_i) = \dot{h}(\mathbf{n}_i), \qquad i \in 0, 1 \ .$$

Suppose, that the corresponding curves $\mathbf{x}_h$, $\mathbf{x}_g$ are cusp-free on U. Then their Hausdorff distance corresponds to the error in support functions.

$$||\mathbf{x}_h - \mathbf{x}_g||_H = ||h - g||_\infty \ . \tag{13}$$

Proof. Due to boundary conditions and absence of singular points (cusps), the Hausdorff distance is realized by a common normal line to both curves. The distance of the points on this line is equal to the absolute value of the difference of the support functions. For a more formal proof see [18, Proposition 14]. □

The approximation space can have a higher dimension than 6 and the remaining degrees of freedom can be used for minimizing the segment error. The two possible strategies are based on interpolation of some additional data and on minimizing some integral measure, respectively.

As we are using an approximation space (10) of dimension 7, after satisfying (12) for both boundary points, we are left one additional free parameter. In the following example we will use this parameter for interpolation of the support function value at the mid-normal

$$g(t') = \sum_{i=1}^{7} c_i a_i(t') = h\left(\frac{\mathbf{n}_0 + \mathbf{n}_1}{2}\right), \quad \text{for } t' = \arctan\frac{n_{01} + n_{11}}{n_{02} + n_{12}} \tag{14}$$

or alternatively to minimize the L_2 norm of the difference of supports. In this case every c_i is a function of the free parameter e used to minimize the quantity

$$||h(t) - g(t, e)||_\infty \ . \tag{15}$$

5 Algorithm and example

In this section we summarize the previous results in an algorithm for topologically precise approximation of algebraic curves. We also demonstrate this algorithm on an example.

5.1 Algorithm description

Algorithm 1 summarizes the process of determining the topology of an algebraic curve and the subsequent approximation of the curve.

In step 1 the implicit definition $D(h, \mathbf{n}) = 0$ of the support function is obtained by eliminating the variables x, y from (3). In the next step we determine the cusps - equation (**??**), the extremal points - equations (9) and the inflection points using Proposition 3. We get corresponding points in Step 3 from the envelope formula (1). Then we try to connect the points found in Step 3 using rules

Algorithm 1 Topologically accurate approximation of an algebraic curve

Input: Real algebraic curve $\mathcal{C}$ given as a zero set of a bivariate polynomial with rational coefficients $f(x,y) \in \mathbb{Q}[x,y]$

Output: Topologically accurate approximation of the curve $\mathcal{C}$.

1: Determine the support function h of $\mathcal{C}$.
2: Determine the cusps, extremal and inflection points in the implicit support function representation.
3: Find corresponding points on the primary curve.
4: Connect points.
5: Determine the self-intersections.
6: Approximate the support function of the segments by trigonometric polynomials.
7: Use envelope formula to find the approximation of $\mathcal{C}$.

from Section 3.2. If this method fails we use a standard CAD based algorithm or additional information, e.g., the curve could be an offset of a known parametric curve, etc. As we have the connectivity of these points we can in Step 5 recover the self-intersections as the intersections of tangent triangles as shown in Proposition 5. The two steps - the approximation is described in Section 4.2.

5.2 Example

In order to demonstrate all features mentioned above, we will use them on the example of the offset at distance $-\frac{9}{10}$ to the ellipse given as the zero set of the bivariate polynomial $f(x,y) = x^2 + 4y^2 - 4$ and oriented by its outer normal.

Eliminating x and y from the system of equations (3)

$$\begin{aligned} x^2 + 4y^2 &= 4 \ , \\ -8yn_1 + 2xn_2 &= 0 \ , \\ xn_1 + yn_2 &= h \ , \end{aligned}$$

we get the implicit definition of support function of f, $D(h,\mathbf{n}) = h^2 - 4n_1^2 - n_2^2 = 0$. The implicit support function of the offset at distance $-\frac{9}{10}$ is therefore easily evaluated as

$$D(h,\mathbf{n}) = \left(h - \frac{9}{10}\right)^2 - 4n_1^2 - n_2^2 = 0 \ .$$

The condition for cusps given by equation (**??**) becomes

$$h - \frac{30(n_1^2 - n_2^2)(10h-9)^2 + 9000n_1^2n_2^2}{(10h-9)^3} = 0$$

and has the 4 solutions listed in Table 1. We determine the extremal points by solving the equations

$$\left(h - \tfrac{9}{10}\right)^2 - 1 = 0 \quad \text{and} \quad \left(h - \tfrac{9}{10}\right)^2 - 4 = 0 \ .$$

These are also in Table 1.

Table 1. Cusps (C) and extremal points (E) of the offset curve at distance $-\frac{9}{10}$ to the ellipse $x^2 + 4y^2 - 4 = 0$.

	type	h	h'	h''	$\mathbf{n}$	corresponding point
P_1	E	$-\frac{11}{10}$	0	$\frac{3}{2}$	$(1, 0)$	$(-\frac{11}{10}, 0)$
P_2	C	-0.7441	0.9039	0.7441	$(0.7535, 0.6575)$	$(-1.155, 0.1918)$
P_3	E	$-\frac{1}{10}$	0	-3	$(0, -1)$	$(0, \frac{1}{10})$
P_4	C	-0.7441	-0.9039	0.7441	$(-0.7535, 0.6575)$	$(1.155, 0.1918)$
P_5	E	$-\frac{11}{10}$	0	$\frac{3}{2}$	$(-1, 0)$	$(\frac{11}{10}, 0)$
P_6	C	-0.7441	0.9039	0.7441	$(-0.7535, -0.6575)$	$(1.155, -0.1918)$
P_7	E	$-\frac{1}{10}$	0	-3	$(0, 1)$	$(0, -\frac{1}{10})$
P_8	C	-0.7441	-0.9039	0.7441	$(0.7535, -0.6575)$	$(-1.155, -0.1918)$

These 8 points $P_1, P_2, \ldots, P_8$ divide the curve into 8 segments. The connectivity is found using rules from Section 3.2. We need only the rule 2, the value of h'' at P_1 is positive and therefore it have to be connected to points on the left from it - there are only two points P_2, P_8. Similarly P_5 is connected to P_4 and P_6. The value of h'' at P_3 is negative and therefore it is connected to points below it, i.e. P_2, P_4. And the same argument is used to connect P_7 to P_6 and P_8. The connectivity is on Fig. 2, left.

For simplicity we use the approximation space of dimension 6

$$\mathcal{A} = \text{Span}\{\sin t, \cos t, \sin 2t, \cos 2t, \sin 3t, \cos 3t\} \ .$$

Solving the system of linear equation (11) we interpolate every arc of the offset by an arc of trigonometric polynomial of degree 3. The resulting spline is on Fig. 2, right.

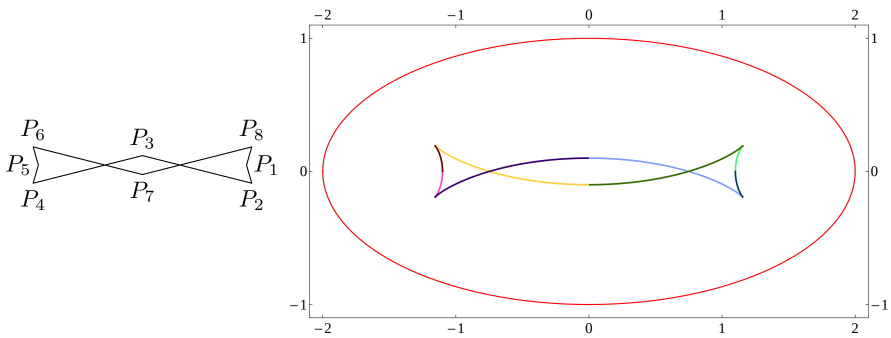

Fig. 2. Left: The graph topologically equivalent to the offset at distance $-\frac{9}{10}$ to the ellipse $x^2 + 4y^2 - 4 = 0$. Right: Its approximation by a spline curve composed of 8 arcs of trigonometric polynomials of degree 3.

Table 2 shows the approximation error and its improvement (ratio of two consecutive errors). The error was obtained by sampling the Hausdorff distance, which is, due to Proposition 7, the maximal difference between the support functions. From the table it seems that the improvement of the error converge to 64, i.e., the approximation order is 6. The graphs of error for first few interpolation degrees are shown in Fig. 3.

Table 2. Errors of the interpolation of offset at distance $-\frac{9}{10}$ to the ellipse $x^2+4y^2-4=0$ by trigonometric spline coming as a solution of (11).

parts	error	improvement
8	$2.43023 \cdot 10^{-3}$	
16	$4.42354 \cdot 10^{-5}$	54.93871
32	$1.26347 \cdot 10^{-6}$	35.01110
64	$3.66130 \cdot 10^{-8}$	34.50865
128	$6.68349 \cdot 10^{-10}$	54.78136
256	$1.08374 \cdot 10^{-11}$	61.67045
512	$1.71052 \cdot 10^{-13}$	63.35748
1024	$2.64063 \cdot 10^{-15}$	64.77699
2048	$4.16170 \cdot 10^{-17}$	63.45069
4096	$6.48248 \cdot 10^{-19}$	64.19919

When we use the approximation space (10) of dimension 7 and use the last degree of freedom to interpolate the support function at mid-normal (condition (14)), the approximation error for 8 segments will decrease cca. 10 times (from $2.43023 \cdot 10^{-3}$ to $2.12534 \cdot 10^{-4}$). The graph of the approximation error is in Fig. 4, left.

We get very similar result when the degree of freedom is used to minimize the L_2 norm of the difference of the support functions, see (15). The optimal values of the parameters are $e_1 = e_4 = e_5 = e_8 = 2.9805$ and $e_2 = e_3 = e_6 = e_7 = -18.6308$, where the index denotes the number of the segment. The approximation error is $2.03011 \cdot 10^{-4}$ and the graph is shown in Fig. 4, right.

Every arc of the offset curve is enclosed in the tangent triangle due to Proposition 4. Therefore the curve yields a self-intersection only if there is a pair of triangles which have an intersection in way described in Proposition 5. From Fig. 5 we see that there are only two self-intersections and we also know their approximate position in the colored polygons. Using all this information we can construct the topologically equivalent graph to the given curve.

6 Conclusion

We have suggested a new approach to the problem of determining the topology of algebraic curves and their approximation. We were systematically using the implicit support function representation of planar curves which is a kind of dual

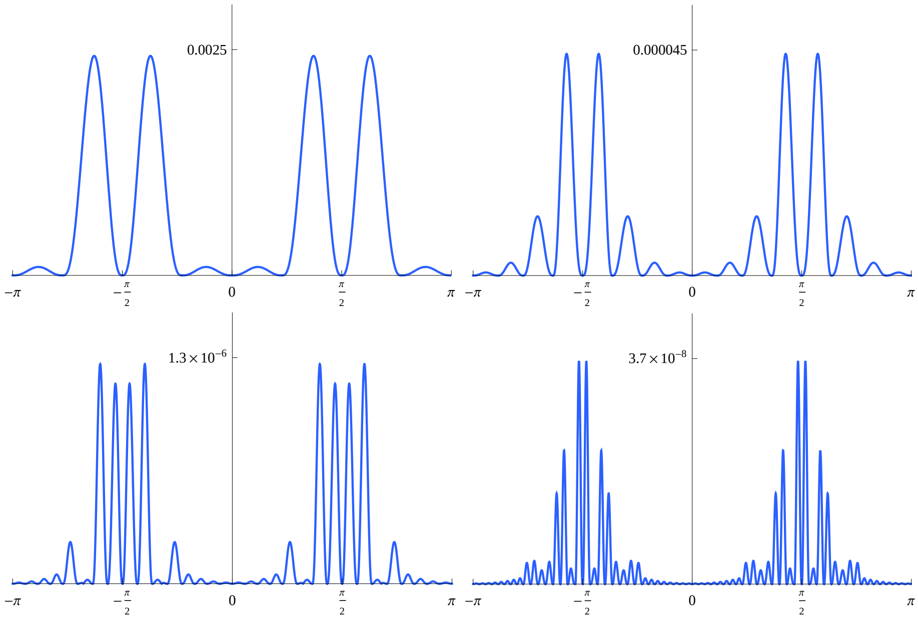

Fig. 3. The approximation error for 8, 16, 32 and 64 segments of spline in approximation space of dimension 6. The points where the error vanishes are the points in which we interpolate the curve.

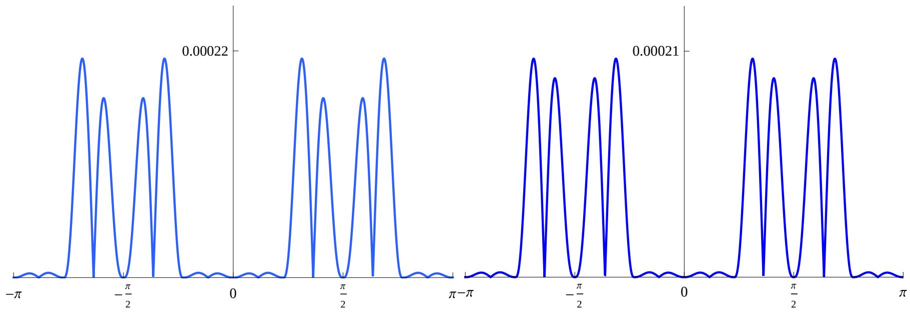

Fig. 4. The graph of the approximation error for 8 segments for different methods of fixing the degree of freedom: left the interpolation of support function at mid-normal, right: the minimization of the L_2 norm of the difference of the support functions.

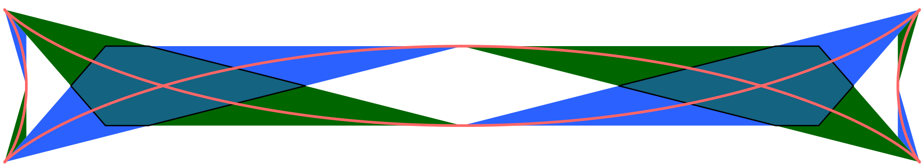

Fig. 5. Every piece of the curve lies inside the envelope triangle. The self-intersections lies inside the intersection of these triangles.

representation. We have illustrated several advantages related in particular to the calculations of cusps, extremal points and inflection points. We also designed a cusp-preserving approximation scheme for regular curve segments.

In the future, we intend to develop the support function based treatment of self-intersections (via an iterative bounding of the area they can occur) and of inflections (in particular their interpolation with suitable multivalued support functions). We also plan to combine our dual techniques with direct computations with primary curve in order to obtain a highly efficient algorithm.

7 Acknowledgement

The first author was supported by the grant of Czech Science Foundation GACR 201/09/H012 and grants of Charles University Grant Agency GAUK 640212 and SVV-2013-267317.

References

1. Aigner, M., Jüttler, B., Gonzalez-Vega, L. , Schicho, J.: Parametrizing surfaces with certain special support functions, including offsets of quadrics and rationally supported surfaces. Journal of Symbolic Computation 44, 180–191 (2009).
2. Aigner, M., Gonzalez-Vega, L., Jüttler, B., Sampoli, M.L.: Computing isophotes on free-form surfaces based on support function approximation. In: Hancock, E., Martin, R., (eds.), The Mathematics of Surfaces (MoS XIII 2009), LNCS, vol. 5654, pp. 1–18. Springer (2009).
3. Bastl, B., Lávička, M., Šír, Z.: G^2 Hermite Interpolation with Curves Represented by Multi-valued Trigonometric Support Functions. LNCS, vol. 6920, 142–156 (2012).
4. Alberti, L., Mourrain, B., Wintz, J.: Topology and arrangements computation of semi-algebraic planar curves. Computer Aided Geometric Design 25, 631–651, (2008).
5. Basu, S., Pollack, R., Roy M.-F.: Algorithms in Real Algebraic Geometry. Springer-Verlag (2006).
6. Eigenwilling, A., Kerber, M. and Wolpert, N.: Fast and Exact Geometric Analysis of Real Algebraic Plane Curves. In: Brown, C.W. (ed.): Proceedings of the 2007 International Symposium on Symbolic and Algebraic Computation (ISSAC 2007), pp. 151–158. ACM (2007).
7. Gonzalez-Vega, L., Necula, I.: Efficient topology determination of implicitly defined algebraic plane curves. Computer Aided Geometric Design 19, 719–749 (2002).
8. Gravesen, J.: Surfaces parametrised by the normals. Computing 79, 175–183 (2007).
9. Gravesen, J., Jütter, B., Šír, Z.: Approximating Offsets of Surfaces by using the Support Function Representation. In: Bonilla, L.L., Moscoso, M., Platero, G., Vega, J.M. (eds.): Progress in Industrial Mathematics at ECMI 2006, Mathematics in Industry 12, pp. 719-723. Springer Verlag (2007).
10. Gravesen, J., Jüttler, B., Šír, Z.: On rationally supported surfaces. Comput. Aided Geom. Design 5 (4-5), 320–331 (2008).
11. Gruber, P.M., Wills, J.M.: Handbook of convex geometry. North–Holland, Amsterdam (1993).

12. Hong H.: An efficient method for analyzing the topology of plane real algebraic curves. Mathematics and Computers in Simulation 42, 571–582 (1996).
13. Cheng, J., Lazard, S., Peñaranda, L., Pouget, M., Rouillier, F., Tsigaridas, E.: On the Topology of Planar Algebraic Curves. Proceedings of the 25th Annual Symposium on Computational Geometry, pp. 361–370. ACM, New York (2009).
14. Lávička, M., Bastl, B., Šír, Z.: Reparameterization of curves and surfaces with respect to convolutions. In: Dæhlen, M. et al. (eds.): MMCS 2008, LNCS, vol. 5862, pp. 285–298, Springer-Verlag Berlin, Heidelberg (2010).
15. Sabin, M.: A Class of Surfaces Closed under Five Important Geometric Operations. Technical Report VTO/MS/207, British Aircraft Corporation (1974). Available at `http://www.damtp.cam.ac.uk/user/na/people/Malcolm/vtoms/vtos.html`
16. Seidel, R., Wolpert, N.: On the exact computation of the topology of real algebraic curves. Proceedings of 21st Annual Symposium on Computational Geometry, pp. 107–115, ACM, New York (2005).
17. Šír, Z., Bastl, B., Lávička, M.: Hermite interpolation by hypocycloids and epicycloids with rational offsets. Computer Aided Geometric Design 27, 405–417 (2010).
18. Šír, Z., Gravesen, J., Jüttler, B.: Curves and surfaces represented by polynomial support functions. Theoretical Computer Science 392, 141–157 (2008).
19. Walker R.J.: Algebraic Curves. Springer-Verlag (1978).