Advances in Computational Mathematics (2007) 27: 1–26

Guaranteed consistency of surface intersections and trimmed surfaces using a coupled topology resolution and domain decomposition scheme

Joel Hass [a], Rida T. Farouki [b], Chang Yong Han [b], Xiaowen Song [c] and Thomas W. Sederberg [d]

[a] *Department of Mathematics, University of California, Davis, CA 95616, USA*
E-mail: hass@math.ucdavis.edu
[b] *Department of Mechanical and Aeronautical Engineering, University of California, Davis, CA 95616, USA*
E-mail: {farouki,cyhan}@ucdavis.edu
[c] *College of Mechanical and Energy Engineering, Zhejiang University, Hangzhou, China*
E-mail: songxw@zju.edu.cn
[d] *Department of Computer Science, Brigham Young University, Provo, UT 84602, USA*
E-mail: tom@cs.byu.edu

Received 7 May 2004; accepted 30 January 2005; published online 9 August 2006
Communicated by T.N.T. Goodman

We describe a method that serves to simultaneously determine the topological configuration of the intersection curve of two parametric surfaces and generate compatible decompositions of their parameter domains, that are amenable to the application of existing perturbation schemes ensuring exact topological consistency of the trimmed surface representations. To illustrate this method, we begin with the simpler problem of topology resolution for a planar algebraic curve $F(x, y) = 0$ in a given domain, and then extend concepts developed in this context to address the intersection of two tensor-product parametric surfaces $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ defined on $(s, t) \in [0, 1]^2$ and $(u, v) \in [0, 1]^2$. The algorithms assume the ability to compute, to any specified precision, the real solutions of systems of polynomial equations in at most four variables within rectangular domains, and proofs for the correctness of the algorithms under this assumption are given.

Keywords: curve topology, ambient isotopy, tensor-product surfaces, surface intersections, trimmed surfaces, surface perturbations, topological consistency, domain decomposition

Mathematics subject classification (2000): 65D17

1. Introduction

The problem of guaranteeing exact agreement of two trimmed surface patches along their common edge, defined by the intersection curve of the original (untrimmed) patches, is of critical importance in computer-aided design and manufacturing [10]. Al-

though the intersections of rational surfaces do not, in general, admit exact rational parameterizations, it is nevertheless possible to impose small perturbations on the original surfaces in such a manner that certain rational approximations of the intersection curve *are* exact for these perturbed surfaces. Two such perturbation schemes for achieving exact "topological consistency" of trimmed surface representations were described in [24] and [12]. In the former, a method is proposed by which the intersection is approximated in the parameter domains of the two surfaces, and surface perturbations are determined so as to ensure an exact match of the images in $\mathbb{R}^3$ of these parameter-domain approximations. In the latter scheme, the intersection is directly approximated as a parametric curve in $\mathbb{R}^3$, and the surface trimming scheme employs triangular patches incorporating this curve as one edge. In both schemes, provisions are made to ensure an appropriate degree of continuity of the trimmed patches with untrimmed patches of the original surface, along their common boundaries.

The topologically-consistent trimmed surface algorithms proposed in [24] and [12] consider pairs of rectangular patches that intersect along a single diagonal arc. In order to accommodate more general surface intersections, these algorithms require a pre-process step in which the intersection curve of the original surfaces is dissected into "simple" monotone segments. Also, the parameter domains of the two surfaces must be subdivided so that each pair of corresponding subpatches intersect in a specified simple manner. The goal of this paper is to give a detailed treatment of the pre-processing phase, in which the topology of the intersection curve is resolved through a domain decomposition that facilitates subsequent application of the surface perturbation scheme to achieve topological consistency.

Our focus in this paper is on methods that give topological descriptions of intersection curves, and dissect them into a collection of "elementary" smooth segments amenable to accurate polynomial or rational approximation. By the methods described in [24] and [12], we may then impose perturbations on the surfaces in the vicinity of each intersection segment, to ensure "water-tight" representations for the trimmed surfaces they define. We seek methods that are mathematically precise and provably correct. The archetypal context for our algorithms is the intersection of two tensor-product polynomial Bézier patches, but the methods can be readily adapted to accommodate triangular patches, B-spline surfaces, rational surfaces, etc.

We present a series of algorithms that analyze the zero sets of polynomial functions, and the intersection curves of two surfaces in $\mathbb{R}^3$. We rigorously prove that these algorithms output topologically correct curve descriptions, given stated assumptions concerning the ability to solve lower-dimensional intersection problems (in particular, to compute the discrete real solutions of certain polynomial equations). The *curve description algorithm* characterizes the zero set of a bivariate polynomial $F(x, y)$ in a rectangular domain. The *intersecting surfaces algorithm* describes the pre-images of the intersection of two parametric surfaces $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ defined on rectangular domains $(s, t) \in P$ and $(u, v) \in Q$. Finally, using the *coordinated domain decompostion algorithm* we construct a subdivision of the two rectangles guaranteeing that the image of each subpatch of P intersects the image of at most one subpatch of Q, and does so in

a "simple" manner. The final output is a set of paired rectangular subpatches of the original surfaces, each pair exhibiting the property that their mutual intersection is a smooth, monotone segment traversing these subpatches between diagonally opposite corners.

Our plan for this paper is as follows. In section 2 we briefly review prior research on the topological analysis of implicitly-defined curves, as background for the algorithms described herein. The problem of characterizing the planar curve described by a polynomial equation $F(x, y) = 0$ within a rectangular domain* is then considered in section 3. This leads to the formulation of the *curve description algorithm* in section 4. In section 5 we extend this algorithm to the context of greatest practical interest – namely, the intersection of two parametric surfaces. It is essential, in this context, to decompose the surfaces into (non-overlapping) paired rectangular subdomains if the surface perturbation schemes of [12,24] are used to ensure exact agreement in $\mathbb{R}^3$ of the intersection approximations, and this requirement is addressed in section 6. Finally, in section 7 we present a computed example to illustrate the working of the algorithm, and in section 8 we summarize our results and make concluding remarks on their practical use.

2. Topological analysis of curves

Although the topological characterization of "implicitly-defined" curves has, for many years, been recognized as a fundamental problem in computational geometry, computer-aided geometric design, computer graphics, and related fields, the research literature that directly addresses this problem is (perhaps on account of its intrinsic difficulty) rather sparse. An algebraic curve in $\mathbb{R}^2$ is specified implicitly as the zero set of a bivariate polynomial, whereas in $\mathbb{R}^3$ it is specified as the intersection of either two implicit surfaces, or two parametric surfaces. Starting from such specifications, the topology analysis must derive the most basic shape information about the curve – such as the number and nature of its components, and their spatial relationships.

Algorithms to perform topological analyses necessarily involve both *logical* and *computational* or "numerical" aspects. For the latter, one may envisage either the use of infallible (but costly) "exact arithmetic" methods involving algebraic field extensions, as commonly used in computer algebra systems, or finite-precision (floating-point) arithmetic – the usual medium for practical applications. Another approach is *variable-precision arithmetic*, where the number of digits is increased "on-the-fly" so as to ensure that the numerical calculations yield consistent logical decisions. Our emphasis in this paper will be on logical aspects of the curve topology analysis, under the assumption that methods are available to compute the real roots of certain polynomial equations to any desired accuracy.

Arnon and McCallum describe a polynomial-time algorithm to determine the topological type of a real (unbounded) plane algebraic curve [2,5] using *cylindrical algebraic decomposition* [3,4]. Assuming a polynomial equation $F(x, y) = 0$ with integer coef-

* This can be considered as a special case of the intersection of two surfaces, where one surface is the graph of a function defined over a plane as the other surface.

ficients, this method relies exclusively on exact-arithmetic computations (an algebraic number, for example, is represented by a rational isolating interval and minimal polynomial, rather than a numerical approximation). Although this algorithm is infallible, the computational cost may grow at an alarming rate as the degree of $F(x, y)$ increases.

Gonzalez-Vega and Necula [15] have recently proposed a "semi-numerical" scheme, which makes use of a computer algebra system to perform a symbolic preprocessing of the curve prior to any numerical calculations. Related work on symbolic and semi-numerical approaches for the determination of curve topology may be found in [1,7,17,20,21].

In most applications, the curve under consideration is restricted to a finite (usually rectangular) domain, and the manner in which it enters and leaves this domain is a key part of the specification of its topological configuration. To ensure that at least one point is found on each component of the curve, the identification of "characteristic points" has been proposed [9]: these include* *border points* (where the curve crosses the domain boundary); *turning points* (where the tangent is horizontal or vertical); and *singular points* (where the curve does not have a unique tangent). However, the use of "curve tracing" [6] to ascertain how these points are connected can incur topological errors.

Grandine and Klein [16] have presented a topology resolution scheme that is similar in many respects to the algorithms described herein. The domain is subdivided into a set of parallel "panels" delineated by the curve singular points and turning points with respect to a prescribed direction. The panel boundaries dissect the curve into segments that are monotone with respect to the chosen direction, and by suitable logic one can identify which of the boundary points are actually connected by curve segments. Our algorithms extend this scheme by ensuring that the curve segments are monotone with respect to both directions, and in the case of intersecting parametric surfaces they guarantee a one-to-one correspondence of the rectangular subdomains containing the monotone segments. Proofs that the algorithms achieve these goals, under stated assumptions, are included below.

The topology algorithm presented in sections 5 and 6 of this paper is motivated and guided by specific requirements of the surface perturbation schemes described in [12,24]. Correct topology is not *per se* sufficient to guarantee a "robust" surface intersection/trimming procedure – one must also ensure consistency of different approximations of the intersection curve segments. The methods described in [12,24] achieve this by applying suitable (linear) perturbations to the surface control points. These methods impose additional requirements on the curve subdivision scheme used in the topology resolution. Specifically, we need a one-to-one correspondence of non-overlapping subsets of the surface parameter domains, that identify smooth intersection segments (see section 6).

* Finding such points involves computing the isolated roots of polynomial systems, and can be considered the 0-dimensional analog of the 1-dimensional problem studied herein.

3. Zero set of a polynomial in a rectangle

We first present a curve description algorithm, that characterizes the zero set of a real polynomial in two variables $F(x, y)$ within a rectangle R. Such sets generically comprise a collection of disjoint arcs, but the algorithm can also accommodate non-generic cases that exhibit singular points. This problem involves finding a certain graph in the plane. To solve it, we assume that we have a method of solving corresponding lower-dimensional problems.

The algorithm input is a polynomial $F(x, y)$ specifying an implicit curve

$$\alpha = \big\{(x, y):\ F(x, y) = 0\big\}, \tag{1}$$

on a rectangular domain

$$R = \big\{(x, y):\ a \leqslant x \leqslant b,\ c \leqslant y \leqslant d\big\}. \tag{2}$$

We allow for the possibility that α may have several components and isolated singularities. The algorithm must accurately characterize the topology of α within R – it will output a collection of vertices and edges, describing a piecewise-linear graph β that is isotopic* to α (i.e., β can be transformed into α by a continuous deformation that fixes the boundary ∂R of R). This guarantees topological correctness, so that α and β have the same number of components and spatial relationships, such as containment and relative location of components. Furthermore, α and β have the same basic geometric features, including the number and location of their maxima and minima. This property will be described precisely in theorem 4.1 below.

We call a point *horizontal* or *vertical* if it lies on α and has a horizontal or vertical tangent line, respectively. Such points comprise the *turning points* of the curve. Turning points are the real solutions within R to the system of two polynomial equations in two unknowns (x, y) defined by

$$F(x, y) = 0$$

and one of

$$F_x(x, y) = 0 \quad \text{or} \quad F_y(x, y) = 0.$$

Singular points satisfy all three of these equations. Such points (if isolated) may be found by standard root-finding procedures [18,23] for polynomials described in the numerically-stable Bernstein representation [11,13,14].

Specifically, the curve description algorithm assumes the following:

1. The zero set of $F(x, y)$ is one-dimensional, with isolated singular points.
2. $F(x, y)$ contains no linear factors of the form $x - x_0$ or $y - y_0$.

* It should be understood that, whenever we speak of an *isotopy* between two curves or graphs, we are referring to an *ambient isotopy* – a continuous family of homeomorphisms of some domain R containing the curves or graphs that fixes the domain boundary ∂R.

3. There exists a method to compute the real zeros of $F(x, y)$ for any specified value $x = x_0$ or $y = y_0$ of either variable.

4. There exists a method to compute the (isolated) intersections of the zero sets of real polynomials $G(x, y)$ and $H(x, y)$ in the rectangle R.

Condition (1) disqualifies polynomials with 2-dimensional zero sets, such as $xy - yx = 0$, or non-isolated singular points, such as $(x - y)^2 = 0$. However, isolated singular points, such as those of $x^2 + y^2 = 0$ or $xy = 0$, are allowed. Condition (2) ensures that no problems arise in using vertical and horizontal subdivisions of the domain R to resolve the curve topology [16]. Condition (3) allows us to find intersections of the curve with the domain boundary, or any horizontal/vertical line. Finally, by applying (4) to a polynomial and its partial derivatives, we can find all turning and singular points of the curve.

The justification for these assumptions, which can be regarded as simpler lower-dimensional versions of the main problem, is as follows (the rigor of the algorithm is predicated on rigorous methods for these subproblems). We can verify condition (1) by checking that the system $F(x, y) = F_x(x, y) = F_y(x, y) = 0$ has finitely many solutions. Condition (2) involves identifying and removing any linear factors of $F(x, y)$ that depend on only one variable – this can be accomplished using univariate polynomial gcd and root-finding procedures (alternately, as in [16], one may invoke a rotation to ensure that linear factors do not define horizontal or vertical lines). Finally, standard methods based upon the subdivision/variation-diminishing properties of the Bernstein polynomial form [14,18,23] can be used to satisfy conditions (3) and (4). In principle, these various requirements can also be implemented in exact symbolic computation for ultimate robustness.

The curve description algorithm is actually applicable in a more general setting, and can be used to determine the topology and key geometry features of any one-dimensional piecewise-analytic set within a rectangle, as long as we can address the lower-dimensional problems enumerated above.

4. Curve description algorithm

A typical curve α defined by the zero set (1) of a polynomial $F(x, y)$ within the domain (2) is illustrated in figure 1. We now describe an algorithm that determines the topological configuration of such curves, and give a proof of its correctness under the stated assumptions. This algorithm can actually accommodate any real analytic curve (having only isolated singular points and a finite number of intersections with any straight line), but for simplicity we focus here on the case of polynomial curves.

Curve description algorithm.

1. *Find all the characteristic (border, turning, and singular) points of* α.
   Figure 1 shows these points for the example curve.

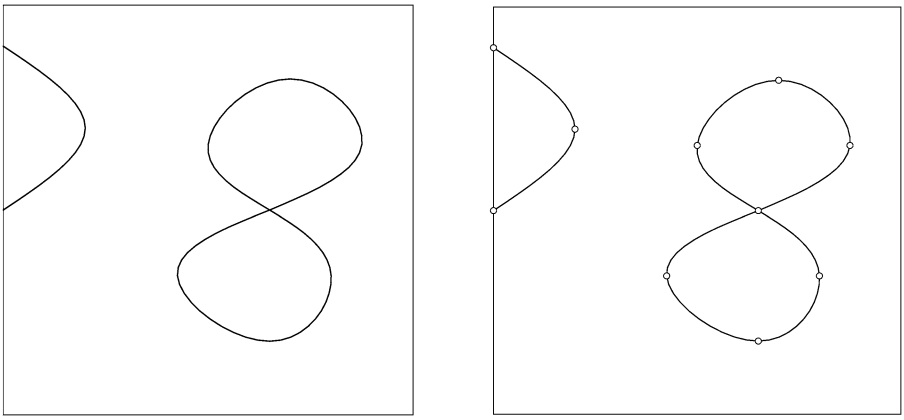

Figure 1. Left: a typical zero set α for the polynomial $F(x, y)$. Right: the set of characteristic (border, turning, and singular) points for this curve.

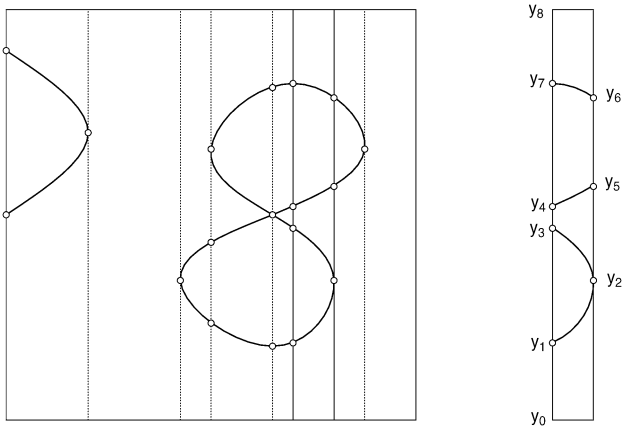

Figure 2. Division of the domain R into vertical strips R_i that have no interior characteristic points. The selected strip R_6 is illustrated on the right.

2. *Divide R into vertical strips without interior characteristic points.*
If $x_1, x_2, \ldots, x_N$ is the ordered sequence of the *distinct* x coordinates of all turning points and singular points of α with $0 < x < 1$, dissect R into $N + 1$ rectangular strips $R_1, R_2, \ldots, R_{N+1}$ along the vertical lines $x = x_i$, $i = 1, \ldots, N$ (note that some turning or singular points may have coincident x coordinates). Find the additional intersection points of α with these vertical lines, as shown in figure 2.

   Each vertical strip may have points of α on its left and right boundary, but their connectivity is not currently known – there may be several different ways to connect them with monotone arcs (see figure 3). The correct connectivity is determined in the following step.

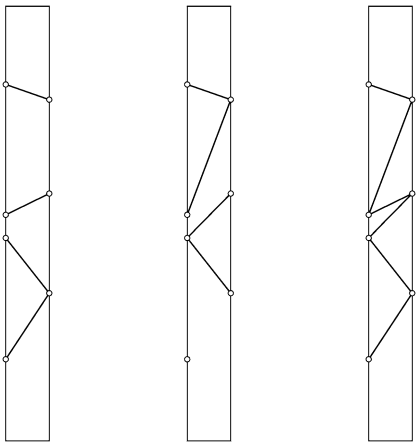

Figure 3. There may be several different ways to connect points of α on the left and right boundaries of a vertical strip, but only one of them is correct.

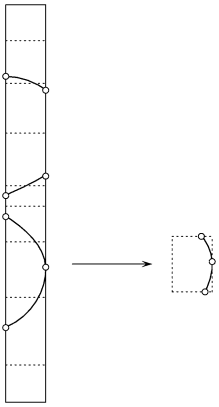

Figure 4. Separating the points of α on the left and right boundaries of each vertical strip R_i by further subdividing with horizontal lines between them.

3. *Determine connectivity of points on boundary of each vertical strip.*
The following process is repeated for each strip R_i, $i = 1, \ldots, N+1$.

Let $c = y_0 < y_1 < \cdots < y_{n_i} < y_{n_i+1} = d$ be the ordered sequence of *distinct* y coordinates of points of α on the interior of the left and right sides of R_i, at $x = x_i$ and x_{i+1}, augmented by the y coordinates of R.

Subdivide the strip R_i by the horizontal lines $y = h_j = \frac{1}{2}(y_j + y_{j+1})$ for $0 \leqslant j \leqslant n_i$, and let the subrectangle containing y_j be denoted by R_{ij} – see figure 4. Note that, by construction, the left and right sides of R_{ij} each contain at most one point of α.

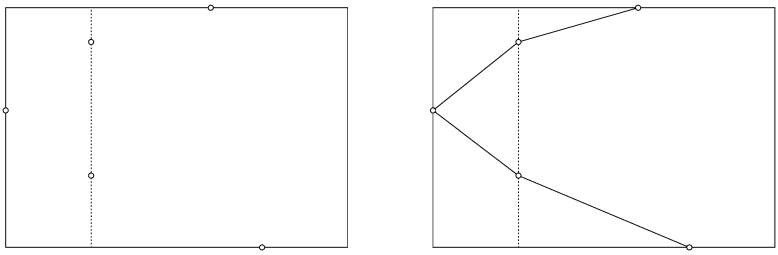

Figure 5. Intersecting with a vertical line to identify the number of increasing and decreasing arcs emanating from a point on the left boundary – this uniquely determines the configuration within the subrectangle.

Compute the intersections of α with each of the horizontal lines $y = h_j$ within the vertical strip, and order them left to right. Let

$$d_{jk} = (x_{jk}, h_j) \quad \text{for } 1 \leqslant k \leqslant m_j$$

be the resulting sequence of points. Also, let the smallest and largest of the coordinates x_{jk} for $1 \leqslant k \leqslant m_j$ and $1 \leqslant j \leqslant n_i$ be $x_{\min}$ and $x_{\max}$.

We inspect each subrectangle R_{ij}, $j = 1, \ldots, n_i$ in turn. If there is a point of α at y_j on the left edge of R_{ij}, compute the intersection points of α in R_{ij} with the vertical line $x = \frac{1}{2}(x_i + x_{\min})$. Let $a_1, a_2, \ldots, a_p$ and $a_{p+1}, a_{p+2}, \ldots, a_{p+q}$ be the ordered y coordinates of these points below and above y_j, respectively. Similarly, if there is a point of α at y_j on the right edge of R_{ij}, we order the y coordinates of the intersections of α with the vertical line $x = \frac{1}{2}(x_{\max} + x_{i+1})$ into two groups $b_1, b_2, \ldots, b_r$ and $b_{r+1}, b_{r+2}, \ldots, b_{r+s}$ – below and above y_j, respectively.

We construct a temporary graph γ_{ij} within R_{ij} as follows:

(a) If there is a point of α at y_j on the left side of R_{ij} and $p > 0$, insert an edge from it to each of the points $d_{j,1}, \ldots, d_{j,p+1}$.

(b) If there is a point of α at y_j on the left side of R_{ij} and $q > 0$, insert an edge from it to each of the points $d_{j+1,1}, \ldots, d_{j+1,q+1}$.

(c) If there is a point of α at y_j on the right side of R_{ij} and $r > 0$, insert an edge from it to each of the points $d_{j,m_j-r+1}, \ldots, d_{i,m_j}$.

(d) If there is a point of α at y_j on the right side of R_{ij} and $s > 0$, insert an edge from it to each of the points $d_{j+1,m_{j+1}-s+1}, \ldots, d_{j+1,m_{j+1}}$.

(e) If $m_j > p + r$, insert edges that connect the $m_j - (p + r)$ pairs of points $(d_{j,p+1}, d_{j+1,q+1}), \ldots, (d_{j,m_j-r}, d_{j+1,m_{j+1}-s})$.

In steps (a)–(d), the edges are drawn through the appropriate points of α on the vertical lines $x = \frac{1}{2}(x_i + x_{\min})$ and $x = \frac{1}{2}(x_{\max} + x_{i+1})$. This process is illustrated in figure 5 for a simple case, and in figure 6 for a more complicated case – where there are points at the same height y_j on the left and right edges of the subrectangle R_{ij}.

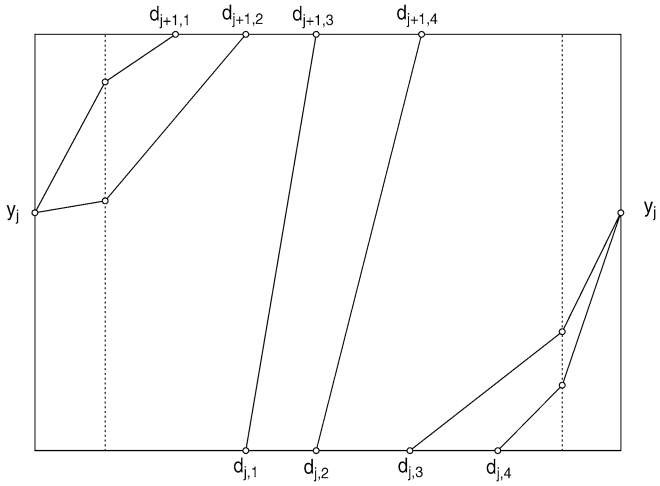

Figure 6. A more complicated case – the curve topology in the subrectangle R_{ij} is still uniquely determined by its boundary points (this is a special case, with boundary points at height y_j on both the left and the right sides of R_{ij}).

Repeating this process on each subrectangle R_{ij} for $j = 1, \ldots, n_i$, we obtain graphs γ_{ij} that consist of non-intersecting polygonal edges with end-points on the boundary ∂R_{ij}. The union of all the subrectangle graphs γ_{ij} for $1 \leqslant j \leqslant n_i$ yields an overall graph γ_i for the vertical strip R_i. This graph γ_i consists of polygonal arcs that are embedded and disjoint in the interior of the strip R_i, and its intersection with each subrectangle R_{ij} consists of polygonal disjoint segments with end-points on ∂R_{ij}. Each edge of γ_i is a sequence of edges in the subrectangles R_{ij} that defines a polygonal arc within R_i, beginning and ending at distinct sides of the boundary of this vertical strip.

We form a new graph β_i for R_i by replacing each polygonal edge of γ_i by the straight-line segment connecting its end-points.*

4. Take the union $\beta = \bigcup_{i=1}^{N+1} \beta_i$ of the graphs for all the vertical strips.
5. Output the resulting set of vertices and connecting edges, along with any isolated vertices meeting no edges, as β.

We now prove the following properties the curve description algorithm:

Theorem 4.1. Assume a method exists to locate all the border, turning, and singular points of the polynomial curve $\alpha = \{(x, y)\colon F(x, y) = 0\}$ for (x, y) in a given rectangle R. Then the *curve description algorithm* constructs a polygonal curve β that is

* By construction, β_i is guaranteed to have the same topological configuration as γ_i. The introduction of intermediate intersection points serves only to ensure correct connectivity of the points of α on the boundary of the strip R_i (theorem 4.1). Preserving the structure of the intermediate graphs yields a more complex, but still topologically accurate, graph.

isotopic to α in R. The isotopy fixes the domain boundary ∂R, and leaves each turning point, border point, and singular point of α fixed. All vertices of β are points on α, and all turning points, border points, and singular points of α are located at vertices of β.

Recall that two curves or graphs in R are *isotopic* (rel ∂R) if there exists a continuous family I_t, $0 \leqslant t \leqslant 1$ of homeomorphisms of R, each fixing the boundary ∂R, with I_0 the identity and I_1 a homeomorphism carrying the first curve to the second. The polygonal curve β captures the correct topology of α, and also captures its key geometric features – the points where it crosses ∂R, where it has a vertical or horizontal tangent, and where it is singular.

The description of β consists of listing the set of vertices $v_1, v_2, \ldots$ and the set of edges $e_i = (v_j, v_k)$ – each edge being a line segment connecting two distinct vertices (there may be some "isolated" vertices, not associated with any edge – for example, isolated singular points of α or points on ∂R not connected to other points of α inside R). If α is nonsingular, it is easy to order the edges so they successively traverse each component of α.

We first give a proof of a fundamental result (well-known to topologists) concerning the topology of arcs in the plane. This result extends the Jordan Curve Theorem to the case of multiple arcs whose interiors are disjoint and embedded in a disk (note that end-points of distinct arcs may coincide).

Lemma 4.2. Let R be a planar region homeomorphic to a closed disk D and let $\gamma = \{\gamma_i\}$, $i = 1, \ldots, n$ be a collection of arcs in R with embedded, disjoint interiors and end-points on ∂R. Let $\gamma' = \{\gamma_i'\}$, $i = 1, \ldots, n$ be a second collection of such arcs in R, γ_i' having the same end-points as γ_i for each i. Then there is an isotopy of R carrying the collection of arcs γ to γ'.

Proof. The Jordan Curve Theorem states that any embedded closed curve in the plane is isotopic to the unit circle. The proof of the theorem also shows that any two embedded arcs in a disk with the same end-points are isotopic by an isotopy that fixes the disk boundary [19]. This establishes the lemma for the case where γ, γ' each comprise a single arc.

If γ has $n > 1$ components $\gamma_1, \ldots, \gamma_n$, lemma 4.2 follows by induction as follows. There is an isotopy F_t of D carrying γ_1 to γ_1' and fixing ∂D. This isotopy carries $\gamma_1, \ldots, \gamma_n$ to arcs $\gamma_1'' = \gamma_1', \gamma_2'', \ldots, \gamma_n''$. The disk D is a union of two regions D_1 and D_2, each homeomorphic to a disk, that have common boundary along the arc $\gamma_1' = \gamma_1''$. The interiors of D_1, D_2 each contain fewer than n of the arcs $\gamma_2'', \ldots, \gamma_n''$. By induction applied to each disk in turn, there is an isotopy F_t' of D carrying the arcs $\gamma_2'', \ldots, \gamma_n''$ to $\gamma_2', \ldots, \gamma_n'$ while leaving the boundaries of D_1, D_2 fixed. The composition of F_t' and F_t gives an isotopy that fixes ∂D and carries $\gamma_1, \ldots, \gamma_n$ to $\gamma_1', \ldots, \gamma_n'$ as desired. □

Proof of theorem 4.1. By assumption, the curve α is a one-dimensional subset of R, and the characteristic points required in step 1 of the algorithm are computable. Since

$F(x, y)$ is of finite degree and has no factors of the form $x - x_0$ or $y - y_0$, the curve α intersects any horizontal or vertical line in finitely many points. In step 2 we subdivide R by a set of vertical lines, breaking the problem of describing the curve α into a collection of subproblems on vertical strips R_i. All turning points and singular points are on the boundaries of these vertical strips, and within their interiors α consists of a set of increasing or decreasing arcs. Each strip R_i is considered in turn.

In step 3 a further subdivision dissects each strip R_i along horizontal line segments, resulting in subrectangles R_{ij}. Each point of α on the left or right side of R_i is separated from the others by a horizontal dividing line, with the possible exception of points on the left and right that have equal y coordinates. Thus, each R_{ij} has at most one point of α on its left or right side, or one point on both sides when these points have identical y coordinates. Since the interior of R_{ij} contains no turning or singular points, the curve α within the interior of R_{ij} consists entirely of arcs, each of which is monotone increasing or decreasing – there are no closed loops or isolated points.

Consider first the case where the left side of R_{ij} contains a point of α at y_j. Arcs of $\alpha \cap R_{ij}$ leaving this point are either increasing or decreasing within R_{ij}. There may be several such arcs if the point at y_j is a turning or singular point of α, and there may be none if it is a turning point or isolated singular point. Any arc leaving this boundary point and entering R_{ij} must intersect the line $x = \frac{1}{2}(x_i + x_{\min})$ in R_{ij}, since in the portion of the subrectangle to the left of this line, there are no intersection points of α with the top or bottom sides, and no turning points. The number of arcs leaving the boundary point is thus equal to the number of intersections of α with this line. The number p of such arcs that are decreasing equals the number of intersection points that lie below y_j, and the number q that are increasing equals the number of intersection points that lie above y_j.

If $p > 0$, the arc joining the left-edge boundary point at y_j to the lowest intersection point of α with the line $x = \frac{1}{2}(x_i + x_{\min})$ must continue descending monotonically until it reaches the bottom side of R_{ij}, since this subrectangle contains no turning points. It meets this side at its left-most intersection point $d_{j,1}$ with α. Continuing arc by arc, we see that there is a unique point of ∂R_{ij} to which every arc emanating from the left-edge boundary point can connect, and as we proceed through the p points of α below y_j on the line $x = \frac{1}{2}(x_i + x_{\min})$, they connect left to right in order to the points $d_{j,1}, \ldots, d_{j,p}$. Likewise, q arcs of α connect the left-edge boundary point to the points $d_{j+1,1}, \ldots, d_{j+1,q}$ ordered left to right on the top side of R_{ij} (see figure 6). If a point of α lies on the right side of R_{ij} at height y_j, a similar argument shows that arcs emanating from it into R_{ij} connect to uniquely-determined points on the upper and lower sides of R_{ij}.

Any remaining "unused" points on the lower and upper sides of ∂R_{ij} must be connected in pairs by monotone arcs of α that run between these sides. In this process, we systematically work from left to right in choosing pairs of points to connect, since these arcs may not cross each other in the interior of R_{ij}. Hence, all points of α on ∂R_{ij} connected by arcs of α can be joined by line segments with disjoint interiors, giving a piecewise-linear graph γ_{ij} in R_{ij}, and by lemma 4.2 there is an isotopy taking $\alpha \cap R_{ij}$ to γ_{ij}.

The algorithm then takes the union of all the graphs γ_{ij} in subrectangles R_{ij} for $1 \leqslant j \leqslant n_i$ comprising the vertical strip R_i, and identifies common vertices in adjacent graphs to form a single graph γ_i in R_i. The graph γ_i is piecewise-linearly embedded in R_i, consisting of a collection of polygonal segments that are either increasing or decreasing. At the end of step 3, these polygonal segments are replaced by line segments connecting their end-points, to form the graph β_i. By a further application of lemma 4.2, there is an isotopy taking γ_i to β_i that fixes the boundary of R_i.

Finally, in step 4 we form an overall graph β for R by taking the union of the graphs β_i for each vertical strip R_i, $i = 1, \ldots, N+1$. Since we can define an overall isotopy of R carrying β to α by taking the union of the isotopies for each vertical strip. The completes the proof. □

5. Intersection of parametric surfaces

Consider now the intersection of tensor-product polynomial* surfaces $\mathbf{p}(s,t)$ and $\mathbf{q}(u,v)$ in $\mathbb{R}^3$, with parameter domains $(s,t) \in P = [0,1]^2$ and $(u,v) \in Q = [0,1]^2$. We assume these surfaces are regular (i.e., $\mathbf{p}_s \times \mathbf{p}_t \neq \mathbf{0}$ and $\mathbf{q}_u \times \mathbf{q}_v \neq \mathbf{0}$) over these domains. We also assume that the two surfaces are embedded – or, equivalently, that $\mathbf{p}(s,t)$ and $\mathbf{q}(u,v)$ define one-to-one maps from P, Q to $\mathbb{R}^3$ (i.e., they do not exhibit self-intersections). We do not assume that they intersect transversely, but do require singular points (where the intersection fails to be transverse), if any, to be isolated.

In formulating an algorithm to resolve the intersection curve topology we assume, as before, a method to solve analogous lower-dimensional problems. Namely, we assume we can find the intersection points of a parametric curve and a parametric surface in $\mathbb{R}^3$. We also assume that we can compute the points corresponding to all turning or singular points of the intersection curve pre-images in the parameter domains P and Q of the two surfaces.

The curve-surface intersection problem corresponds to three equations in three unknowns (that arise from equating coordinate components in $\mathbb{R}^3$): the two surface parameters and the single curve parameter. The identification of turning points corresponds to solving four equations in four unknowns (singular points involve an additional condition). For polynomial surfaces, these problems can be solved by Bernstein-form root-finding methods.

We now describe the *intersecting surfaces algorithm*. This closely parallels the *curve description algorithm* of section 4, although the terminology is different. Theorem 5.1 gives a formal statement of key properties of this algorithm. Let $\mathbf{p}(s,t)$ and $\mathbf{q}(u,v)$ be mappings of the domains $P = [0,1]^2$ and $Q = [0,1]^2$ into $\mathbb{R}^3$, each coordinate component of $\mathbf{p}(s,t)$ and $\mathbf{q}(u,v)$ being a polynomial in the surface parameters. Denote by α_P and α_Q the pre-images in P and Q of the intersection curve α of $\mathbf{p}(s,t)$

* The method can be extended to rational, analytic, or piecewise-analytic surfaces – for brevity, we focus here on polynomial surface patches.

and $\mathbf{q}(u, v)$ in $\mathbb{R}^3$. Assuming that α_P and α_Q have no components* of the form $s = s_0$, $t = t_0$ and $u = u_0$, $v = v_0$ respectively, the algorithm returns two piecewise-linear graphs $\beta_P \in P$ and $\beta_Q \in Q$ that give faithful topological descriptions of α_P and α_Q, as indicated in theorem 5.1 below.

Intersecting surfaces algorithm.

1. *Find all characteristic (border, turning, and singular) points of α_P, α_Q.*
   Border points can be found by intersecting the four boundary curves $\mathbf{p}(s, 0)$, $\mathbf{p}(s, 1)$, $\mathbf{p}(0, t)$, $\mathbf{p}(1, t)$ of $\mathbf{p}(s, t)$ with $\mathbf{q}(u, v)$, and vice-versa (note that border points for α_P may correspond to points in the interior of the domain of α_Q, and vice-versa). The turning and singular points may be found [24] as solutions of four equations in the four variables (s, t, u, v) – namely, the coordinate components of the vector equation $\mathbf{p}(s, t) = \mathbf{q}(u, v)$ and each in turn of the scalar equations $(\mathbf{q}_u \times \mathbf{q}_v) \cdot \mathbf{p}_s = 0$, $(\mathbf{q}_u \times \mathbf{q}_v) \cdot \mathbf{p}_t = 0$ and $(\mathbf{p}_s \times \mathbf{p}_t) \cdot \mathbf{q}_u = 0$, $(\mathbf{p}_s \times \mathbf{p}_t) \cdot \mathbf{q}_v = 0$.
2. *Divide P, Q into vertical strips without interior characteristic points.*
   If $s_1, s_2, \ldots, s_N$ is the ordered sequence of *distinct* s values of all border points, turning points, and singular points of α_P with $0 < s < 1$, dissect P into $N + 1$ rectangular strips $P_1, P_2, \ldots, P_{N+1}$ along the vertical lines $s = s_i$, $i = 1, \ldots, N$, and find the additional intersection points of α_P with these vertical lines. Likewise for α_Q.
3. *Determine connectivity of points on the boundary of each vertical strip.*
   Apply step 3 of the curve description algorithm to construct graphs within each vertical strip of P and Q that describe the connectivity of the exact points of α_P and α_Q on the boundaries of these strips.**
4. Take the union of the graphs over all the vertical strips, to form overall graphs β_P and β_Q in P and Q.
5. Output the resulting sets of vertices and connecting edges as β_P, β_Q.

We prove the following properties of the intersecting surfaces algorithm:

Theorem 5.1. Assume methods exist to compute the intersection points of a parametric curve and a parametric surface in $\mathbb{R}^3$, and to compute all turning points and singular points of the pre-images $\alpha_P \subset P$ and $\alpha_Q \subset Q$ of the intersection of two surfaces $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ defined on $(s, t) \in P = [0, 1]^2$ and $(u, v) \in Q = [0, 1]^2$. Then the *intersecting surfaces algorithm* constructs a pair of piecewise-linear graphs, $\beta_P \subset P$ and $\beta_Q \subset Q$, that are isotopic to α_P and α_Q respectively. The isotopies fix the boundaries of P and Q, and also fix each border point, turning point, and singular point of α_P and α_Q.

* As in section 3, this requirement can be addressed using gcd and root-finding procedures for certain univariate polynomials.

** The logic is identical to that of the curve description algorithm: the only difference is that border points may occur in the interior of the domains P and Q (but not within the vertical strips).

Proof. In the proof of theorem 4.1 for the planar curve case, we found a piecewise-linear graph isotopic to a given curve by identifying border points, turning points, singular points, and intersections with horizontal or vertical lines. In the present context, an additional type of point occurs. It is possible for an arc of α_P or α_Q to terminate at an interior point of P or Q. The location of such points can be found by solving for the intersection of the boundary curves of $\mathbf{p}(s, t)$ with the interior of $\mathbf{q}(u, v)$ and vice-versa. Since we can identify such points on the curves $\alpha_P \subset P$ and $\alpha_Q \subset Q$, following the procedure of theorem 4.1 proves theorem 5.1 by identical arguments. □

Remark. The assumption that the surfaces are embedded can be weakened. It suffices to assume that they are *immersed*. This means that each point of the surfaces has a neighborhood that is embedded, but $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ may have self-intersections: the self-intersections must be tranverse except at isolated points, as with the intersection of $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$. Vertices of α_P that correspond to points in $\mathbb{R}^3$ where $\mathbf{p}(s, t)$ intersects a double curve of $\mathbf{q}(u, v)$ can be located by solving the six equations defined by

$$\mathbf{p}(s, t) = \mathbf{q}(u, v) \quad \text{and} \quad \mathbf{p}(s, t) = \mathbf{q}(u', v')$$

in the six unknowns s, t, u, v, u', v' where $(u, v) \neq (u', v')$. Points where $\mathbf{p}(s, t)$ is tangent to itself and self-intersection points lying along a line $s = s_0$ or $t = t_0$ can be found as before. The algorithm is modified so that, in step 2, additional vertical subdivisions are made along points where α_P or α_Q have singularities in P or Q, respectively.

6. Subdomain correspondence algorithm

The algorithm described in section 5 resolves the topology of the parameter domain pre-images α_P and α_Q of the intersection curve of $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$, and subdivides them into monotone segments. However, to apply the topologically-consistent surface perturbation schemes [12,24] we must further process these representations, to ensure a one-to-one correspondence between monotone segments within rectangular subsets of the parameter domains P and Q of the surfaces. This is achieved by the algorithm described below.

The assumptions in section 5 concerning the intersection of $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ are also invoked for the present algorithm. Figure 7 shows a typical example. We also assume, in the present context, that the intersection is nonsingular. This assumption is motivated by the desire to maintain a succinct algorithm description, bypassing technical difficulties with subdivision in the vicinity of singular points that warrant a separate thorough investigation.

The pre-images α_P and α_Q consist of collections of simple closed loops and open segments. The latter may have end-points lying on the boundaries of the surface domains P and Q, or within their interiors. The algorithm of section 5 constructs polygonal curves β_P and β_Q that are isotopic to α_P and α_Q. The curve β_P is described by a set of vertices $\{p_i = (s_i, t_i)\}$ and a set of edges $\{e_i = (p_j, p_k)\}$ with each edge (p_j, p_k)

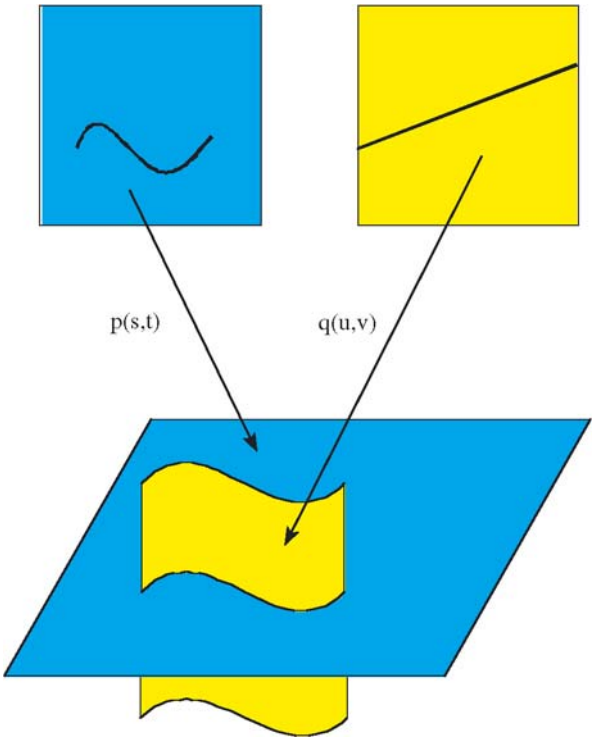

Figure 7. The intersection of $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ has pre-images α_P and α_Q in the respective parameter domains P and Q of these surfaces.

being contained in a subrectangle $P_{jk} \subset P$ with p_j and p_k as diagonally opposite vertices. The subrectangles P_{jk} are non-overlapping and contain only a single intersection curve segment between opposite corners – a collection of such subrectangles is shown in figure 12. The same properties hold for β_Q, with vertices $\{q_i = (u_i, v_i)\}$ and monotone edges $\{e_i = (q_j, q_k)\}$ contained within subrectangles Q_{jk} having vertex pairs q_j and q_k as opposite corners.

However, at this point, the subrectangles $P_{jk} \subset P$ and $Q_{jk} \subset Q$ may not be in one-to-one correspondence, which is essential to achieving topological consistency of trimmed surfaces by the surface perturbation schemes [12,24]. A simple example illustrates this point. In figure 8 we show the intersection curve of figure 7 in the parameter domains P and Q of the surfaces, with characteristic points (border points and turning points) indicated. Figure 9 shows a subdivision of the domains along vertical lines through these points. In figure 10 a further subdivision of the domain P by horizontal lines is shown (the images in Q of the turning points in P are also shown).

Each subrectangle now contains at most one diagonal segment, but these subrectangles are not paired, since there are three in P and just one in Q. To achieve a pairing, we need additional subdivision of Q by vertical and horizontal lines through the images in Q of the turning points in P, and vice-versa (see figure 11). Figure 12 shows

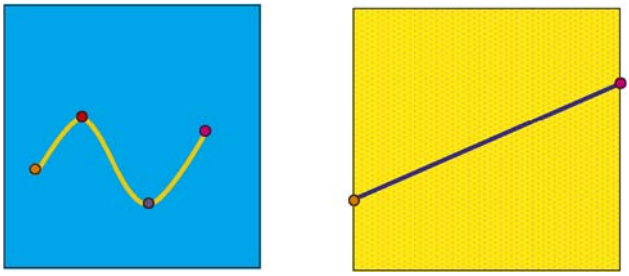

Figure 8. Turning and border points of α_P and α_Q identified.

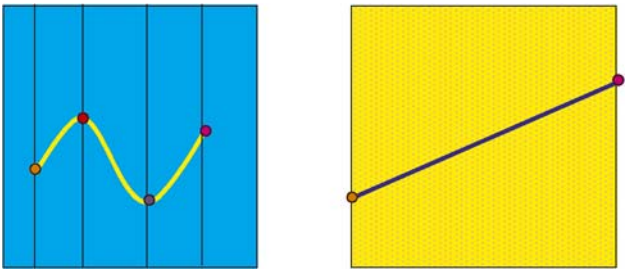

Figure 9. Subdivision along vertical lines through characteristic points.

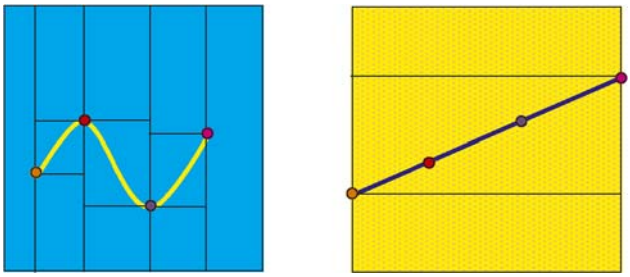

Figure 10. Further subdivision along horizontal lines leads to diagonal arcs in each parameter domain.

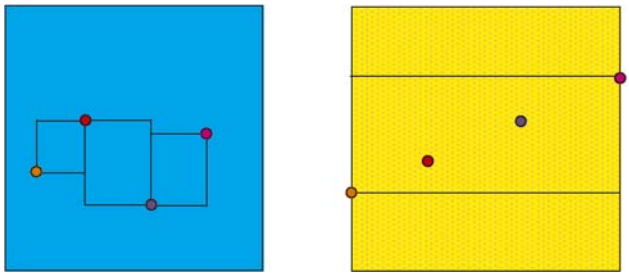

Figure 11. The (u, v) vertices added to (s, t) domain, and vice-versa.

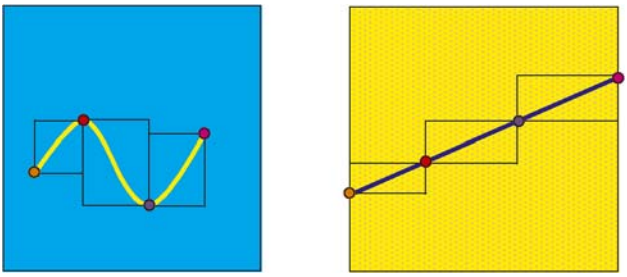

Figure 12. Final rectangles come in pairs, each containing a monotone arc.

the final sets of paired subrectangles, achieved by this additional subdivision. At this point, corresponding intersection segments connecting vertices $\{(p_j, p_k)\}$ and $\{(q_j, q_k)\}$ of these subrectangles have identical images under $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ – as required by the surface perturbation schemes.

We now give a formal description of the algorithm to achieve coordinated decomposition of the surface parameter domains, and establish its properties.

Coordinated domain decomposition algorithm. Let $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ be smooth surfaces defined by vector mappings into $\mathbb{R}^3$ of the domains $P = (s, t) \in [0, 1] \times [0, 1]$ and $Q = (u, v) \in [0, 1] \times [0, 1]$. We assume these surfaces are embedded, and intersect transversely in a nonsingular one-dimensional curve, the image of the curves $\alpha_P \subset P$ under $\mathbf{p}(s, t)$ and $\alpha_Q \subset Q$ under $\mathbf{q}(u, v)$.

1. *Divide P, Q into vertical strips without interior characteristic points.*
   Find all turning points and border points of α_P and α_Q (we assume that there are no singular points), and subdivide P and Q into vertical strips through these points, so that all characteristic points lie on the left or right boundaries of such strips. Note that a border point for α_P may lie in the interior of Q, and vice-versa.

2. *Subdivide strips to obtain rectangles containing diagonal segments.*
   Apply the following process to each vertical strip $s_i \leqslant s \leqslant s_{i+1}$ of P. Draw a horizontal line through each point of α_P on the left side of the strip, and compute the intersections (if any) of these lines with α_P inside the strip. If there are no such intersections, subdivision of the strip is not necessary. Otherwise, let $s_{\min}$ be the smallest s-coordinate of the intersection points. If $s_{\min} < s_{i+1}$, subdivide the vertical strip $s_i \leqslant s \leqslant s_{i+1}$ into the two strips $s_i \leqslant s \leqslant s_{\min}$ and $s_{\min} \leqslant s \leqslant s_{i+1}$. Repeat this process for the strip $s_{\min} \leqslant s \leqslant s_{i+1}$, and keep repeating in this manner until $s_{\min} = s_{i+1}$. Also apply the entire process to each strip for α_Q in the domain Q.

3. *Construct the graph representations β_P, β_Q.*
   Apply step 3 of the curve description algorithm to α_P and α_Q to obtain the piecewise-linear graphs β_P consisting of vertices $\{p_i\}$ and edges $\{(p_j, p_k)\}$, and β_Q consisting of vertices $\{q_i\}$ and edges $\{(q_j, q_k)\}$.

4. *Refine β_P and β_Q to obtain paired subrectangles.*
   For each vertex $p_* = (s_*, t_*)$ of β_P, add a vertex $q_* = (u_*, v_*)$ to β_Q with coordinates satisfying $\mathbf{q}(u_*, v_*) = \mathbf{p}(s_*, t_*)$, and vice-versa. If q_* lies in the interior of the subrectangle Q_{jk} with vertices q_j and q_k as opposite corners, replace the edge (q_j, q_k) with the two edges (q_j, q_*) and (q_*, q_k). The subrectangle Q_{jk} containing a diagonal segment is replaced by two subrectangles containing diagonal segments, with a common vertex at q_*. Similarly, if p_* lies in the interior of subrectangle P_{jk}, replace the edge (p_j, p_k) with the two edges (p_j, p_*) and (p_*, p_k), and replace P_{jk} by two subrectangles containing these edges.

   Once β_P, β_Q have been refined in this manner, the new subrectangles of P, Q containing diagonal edges are in one-to-one correspondence.

5. *Output β_P and β_Q.*
   Output lists of vertices $\{p_i\}$ and $\{q_i\}$ and edges $\{(p_j, p_k)\}$, $\{(q_j, q_k)\}$ defining β_P and β_Q.

The following theorem establishes key attributes of the above algorithm (we again assume the ability to solve the appropriate lower-dimensional root-finding problems).

Theorem 6.1. Suppose $P = [0, 1] \times [0, 1]$ and $Q = [0, 1] \times [0, 1]$ are mapped to $\mathbb{R}^3$ by polynomials $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$, and these surfaces are embedded and intersect transversely. Then the coordinated domain decomposition algorithm yields a pair of polygonal curves $\beta_P \subset P$ and $\beta_Q \subset Q$ satisfying:

1. All turning points and boundary points are at vertices of β_P, β_Q.
2. Each edge (p_j, p_k) of β_P defines a rectangle P_{jk} with p_j, p_k as diagonally opposite vertices, and any two such rectangles have disjoint interiors in P. Similarly, edges (q_j, q_k) of β_Q define rectangles Q_{jk} with q_j, q_k as opposite vertices, and these rectangles have disjoint interiors in Q.
3. β_P is isotopic to α_P in P and β_Q is isotopic to α_Q in Q, by isotopies that fix all turning points and boundary points.
4. There is a one-to-one pairing of vertices $p_i \in \beta_P$ and $q_i \in \beta_Q$, with paired vertices having the same images in $\mathbb{R}^3$ under $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$.
5. The arc of α_P contained in P_{jk} and the arc of α_Q contained in Q_{jk} have the same image in $\mathbb{R}^3$.

Proof. Step 1 of the algorithm subdivides P and Q into vertical strips so that all turning points and border points lie on the left or right sides of these strips. Step 2 further subdivides each strip $s_i \leqslant s \leqslant s_{i+1}$ vertically, so that the subset to the left of the vertical line $s = s_{\min}$ has the following property: any horizontal line through a point of α_P on the left side does not meet α_P in any other point, except possibly at the right side. It follows that if we subdivide the strip $s_i \leqslant s \leqslant s_{\min}$ into subrectangles along the horizontal lines

$t = t_i$ corresponding to the t-coordinates of the points of α_P on the left and right sides, then each of these subrectangles is either empty or contains a segment of α_P without singular or turning points – i.e., a single monotone arc between diagonally opposite corners. The same holds for α_Q.

To establish that the vertical subdivisions in step 2 can occur only finitely many times, note that any horizontal line $t = t_*$ can intersect α_P in only a finite number of points, since α_P is an algebraic curve. The least horizontal distance between two such points at height $t = t_*$ gives a function $d(t_*)$ whose minimum is $d_{\min} > 0$. Each subdivision creates a rectangle whose width is at least $d_{\min}$, so the subdivisions occur finitely often.

Step 3 applies the third step of the curve description algorithm. This constructs a diagonal segment of β_P in each subrectangle if it is non-empty. Lemma 4.2 implies that the diagonal segment connecting the two diagonally opposite vertices is isotopic to the exact intersection curve within the subrectangle. A similar argument applies to β_Q.

In step 4, extra vertices are added to β_P and β_Q. The image of each vertex of β_P in the domain Q is added to β_Q, if it is not already present, and vice-versa. Upon adding a new vertex to β_P within a subrectangle that contains a single monotone arc of α_P, the existing edge of β_P in that subrectangle is replaced by two edges, coincident at the new vertex. The new graph β_P therefore maintains its isotopic relationship to the α_P. After introducing these additional vertices, each vertex of α_P and β_P is paired with a vertex of α_Q and β_Q and the augmented sets of subrectangles P and Q have the following properties. Each contains a monotone arc of α_P and α_Q, and these arcs exhibit one-to-one correspondence, established by the fact that pairs of them have the same images in $\mathbb{R}^3$ under $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$. Since any two diagonals between given corners of a subrectangle are isotopic, the isotopy class of β_P and β_Q is not changed by the vertex additions. All conditions claimed in the theorem are thus satisfied. □

Although the intersecting surfaces algorithm (see section 5) can accommodate singular intersections, we restricted the coordinated domain decomposition algorithm to nonsingular intersections. At singular points with distinct real tangents, it is not always possible to separate the curve segments emanating from such points using just vertical and horizontal lines, as required by our algorithm. This difficulty becomes even more pronounced when the tangents at the singular point are not all distinct, as with a cusp. For such cases, a different strategy is required to dissect the curve into monotone segments, contained within disjoint parameter subdomains that exhibit a one-to-one correspondence between the two surfaces. This is a nontrivial problem, and we defer detailed investigation of it to a subsequent study.

7. Computed example

We illustrate the application of the coupled topology resolution and domain decomposition method in the context of the surfaces $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ shown in figure 13.

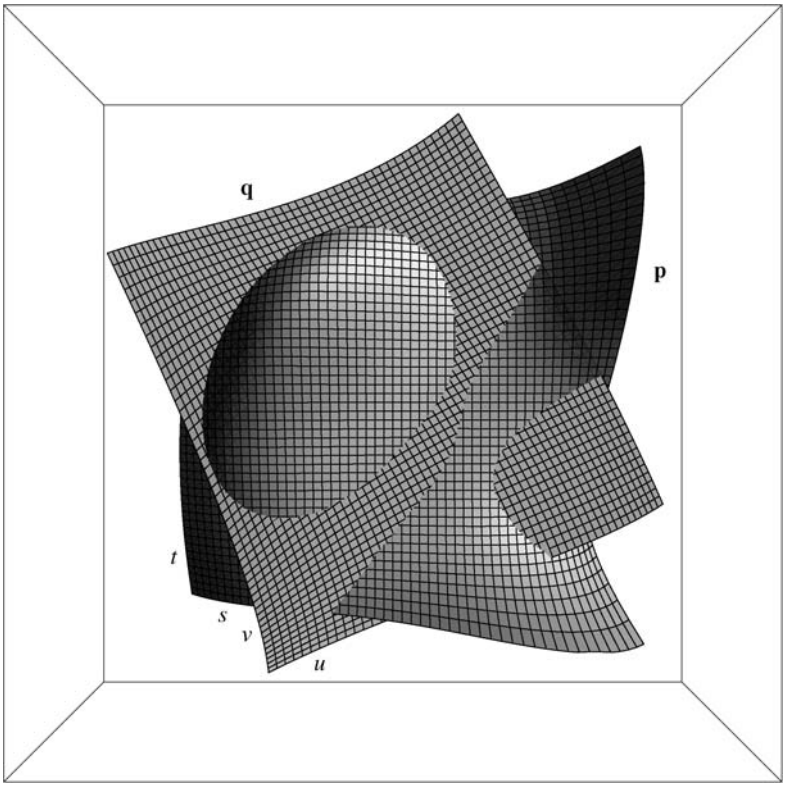

Figure 13. Two parametric polynomial surfaces, $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$, with the directions of the parameters s, t and u, v shown. The topology resolution and domain decomposition scheme is to be applied to their curve of intersection.

These are tensor-product polynomial parametric surfaces, of degree (5, 5) and (3, 3) respectively in the surface parameters. Since a tensor-product surface of degree (m, n) can be implicitized to obtain an equation $f(x, y, z) = 0$ of degree $2mn$ [22], Bezout's theorem indicates that the exact intersection of such surfaces is an algebraic space of degree 900.

Figures 14 and 15 illustrate four successive stages in the progression of the algorithm, within the parameter domains of the surfaces $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$. In these illustrations, the figures in the left columns show particular steps to compute new vertices, while those in the right columns show corresponding updates to the graph structures (the exact intersection loci are also shown in the left-column figures, for ease of reference).

The first rows of these figures show the computation of all characteristic points (border points, turning points, and singular points) of the intersection in the (s, t) and (u, v) planes. Note that some of the border points may lie in the interior or exterior of the parameter domains $(s, t) \in P = [0, 1]^2$ and $(u, v) \in Q = [0, 1]^2$ of the two surfaces.

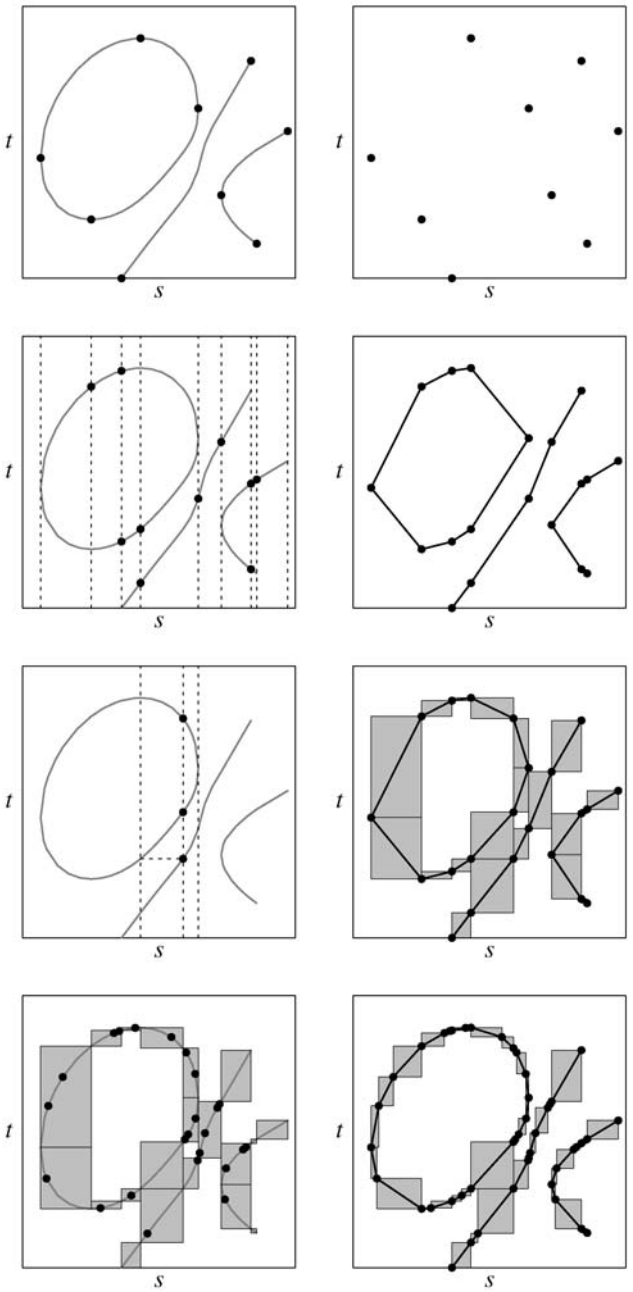

Figure 14. Four stages during the progression of the topology resolution and domain decomposition scheme in the parameter domain of the surface $\mathbf{p}(s, t)$.

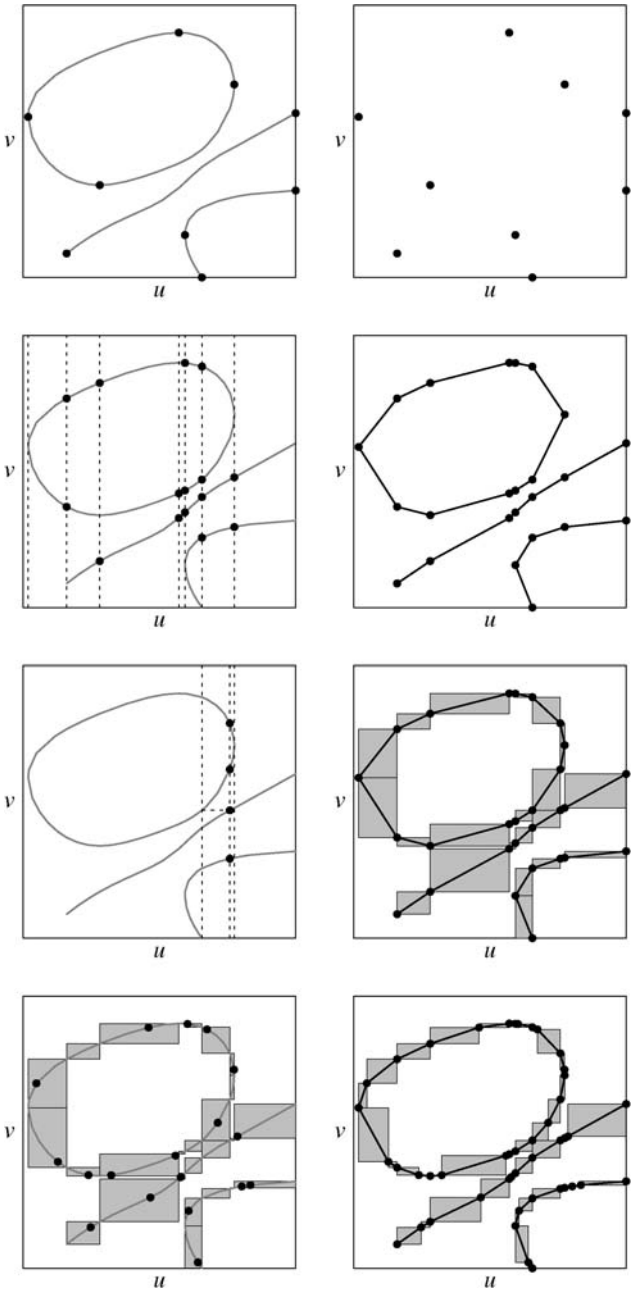

Figure 15. Corresponding stages of the algorithm for the surface $\mathbf{q}(u, v)$.

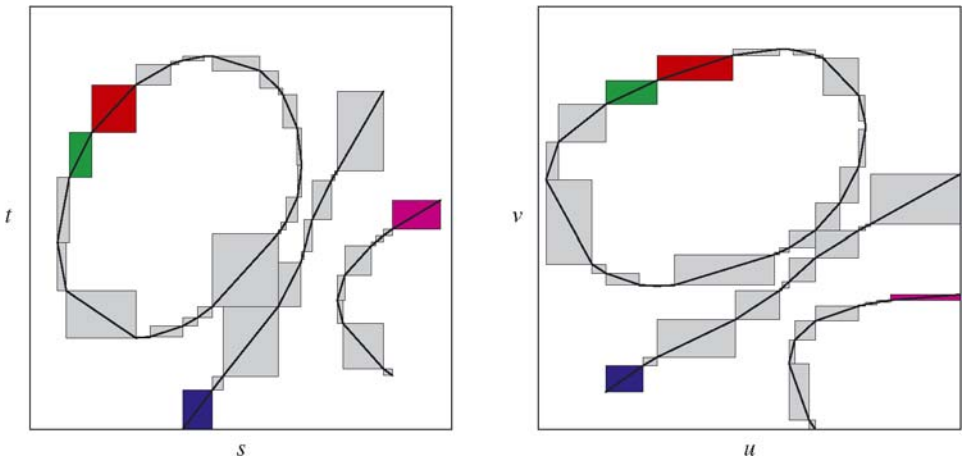

Figure 16. Illustration of the one-to-one correspondence of subdomains of $\mathbf{p}(s,t)$ and $\mathbf{q}(u,v)$ generated by the algorithm. A selection of corresponding subdomains, encompassing intersection segments that are monotone with respect all four surface parameters, are indicated here by color coding.

As shown in the right-hand columns, the connectivity of the characteristic points is not known at this stage.

In the second rows of the figures, vertical lines are drawn through each of the interior characteristic points in the (s,t) and (u,v) domains, and the additional intersections of these lines with the (s,t) and (u,v) pre-images of the intersection are computed (left columns). Once these additional points have been inserted, the graph structures shown in the right-hand figures can be determined by the procedure described in section 5.

The third rows of figures 14 and 15 illustrate step 2 of the coordinated domain decomposition algorithm (see section 6). In this particular example, only one vertical strip requires further subdivision in step 2. In the left columns of these figures, we compute the intersection of the horizontal lines with α_P and α_Q, and then subdivide the vertical strip by the vertical lines passing through the intersection points. In the right columns, we update the graph structure with the additional points thus generated. Note that the subrectangles containing each edge are now all mutually disjoint.

Finally, step 4 of the algorithm is illustrated in the last rows of figures 14 and 15. Each vertex of β_P in the domain P is mapped to the domain Q (if its image is not already present in that domain), and vice-versa. Existing edges in the graph structures β_P and β_Q are replaced by edge pairs wherever necessary, in order to accommodate the introduction of these new vertices. Upon completion of this process, the edges of the refined graph structures β_P and β_Q are in one-to-one correspondence.

In figure 16 we show the final graph structures β_P and β_Q representing the pre-images α_P and α_Q of the intersection curve, together with the parameter subdomains generated by the algorithm of section 6. A selection of corresponding pairs of subdomains are shown color-coded in this figure.

8. Closure

We have described algorithms to resolve the topology of the intersection curve of two polynomial or rational parameteric surfaces $\mathbf{p}(s, t)$ and $\mathbf{q}(u, v)$ and to perform a coordinated decomposition of their parameter domains, such that the rectangular subdomains contain single monotone intersection segments and are in one-to-one correspondence. Such algorithms are prerequisites for application of surface perturbation schemes [12,24] that ensure topological consistency of the trimmed surfaces delineated by the intersection curve, by applying small displacements to the surface control points in its vicinity.

The algorithms assume the ability to compute, to any desired precision, the real roots of univariate polynomials within a given interval, and of pairs of bivariate polynomials within given rectangles. The focus of the algorithm descriptions is on their logical, geometrical, and topological aspects, rather than particular details of the computational model. For practical software implementations in double-precision floating-point arithmetic, a high degree of robustness in the root-solving functions can be achieved by exploiting the subdivision and variation-diminishing properties of the numerically-stable Bernstein representation [11,13,14]. If ultimate robustness is desired, the algorithms could be implemented using symbolic computation, assuming the surfaces are specified exactly by control points with rational coordinates.

Acknowledgements

This work was supported by NSF and DARPA through grant DMS-0138411.

References

[1] S. Arnborg and H. Feng, Algebraic decomposition of regular curves, J. Symbolic Comput. 5 (1988) 131–140.
[2] D.S. Arnon, Topologically reliable display of algebraic curves, ACM Computer Graphics 17 (1983) 219–227.
[3] D.S. Arnon, G.E. Collins and S. McCallum, Cylindrical algebraic decomposition I: The basic algorithm, SIAM J. Comput. 13 (1984) 865–877.
[4] D.S. Arnon, G.E. Collins and S. McCallum, Cylindrical algebraic decomposition II: An adjacency algorithm for the plane, SIAM J. Comput. 13 (1984) 878–889.
[5] D.S. Arnon and S. McCallum, A polynomial-time algorithm for the topological type of a real algebraic curve, J. Symbolic Comput. 5 (1988) 213–236.
[6] C. Bajaj, C.M. Hoffmann, R.E. Lynch and J.E.H. Hopcroft, Tracing surface intersections, Comput. Aided Geom. Design 5 (1988) 285–307.
[7] P. Cellini, P. Gianni and C. Traverso, Algorithms for the shape of semialgebraic sets: A new approach, in: *Lecture Notes in Computer Science*, Vol. 539 (Springer, New York, 1991) pp. 1–18.
[8] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design* (Academic Press, 1997).
[9] R.T. Farouki, The characterization of parametric surface sections, Computer Vision, Graphics, Image Processing 33 (1986) 209–236.

[10] R.T. Farouki, Closing the gap between CAD model and downstream application (Report on the SIAM Workshop on Integration of CAD and CFD, UC Davis, April 12–13, 1999), SIAM News 32 (1999) 1–3.
[11] R.T. Farouki and T.N.T. Goodman, On the optimal stability of the Bernstein basis, Math. Comput. 65 (1996) 1553–1566.
[12] R.T. Farouki, C.Y. Han, J. Hass and T.W. Sederberg, Topologically consistent trimmed surface approximations based on triangular patches, Comput. Aided Geom. Design 21 (2004) 459–478.
[13] R.T. Farouki and V.T. Rajan, On the numerical condition of polynomials in Bernstein form, Comput. Aided Geom. Design 4 (1987) 191–216.
[14] R.T. Farouki and V.T. Rajan, Algorithms for polynomials in Bernstein form, Comput. Aided Geom. Design 5 (1988) 1–26.
[15] L. Gonzalez-Vega and I. Necula, Efficient topology determination of implicitly defined algebraic plane curves, Comput. Aided Geom. Design 19 (2002) 719–743.
[16] T.A. Grandine and F.W. Klein, A new approach to the surface intersection problem, Comput. Aided Geom. Design 14 (1997) 111–134.
[17] H. Hong, An efficient method for analyzing the topology of plane real algebraic curves, Math. Comput. Simulation 42 (1996) 571–582.
[18] J.M. Lane and R.F. Riesenfeld, Bounds on a polynomial, BIT 21 (1981) 112–117.
[19] M.H.A. Newman, *Topology of Plane Sets* (Cambridge Univ. Press, Cambridge, 1939).
[20] M.-F. Roy and A. Szpirglas, Complexity of the computation of cylindrical decomposition and topology of real algebraic curves using Thom's lemma, in: *Lecture Notes in Mathematics*, Vol. 1420 (Springer, New York, 1990) pp. 223–236.
[21] T. Sakkalis, The topological configuration of a real algebraic curve, Bull. Austral. Math. Soc. 43 (1991) 37–50.
[22] T.W. Sederberg, Implicit and parametric curves and surfaces for computer aided geometric design, Ph.D. thesis, Purdue University (1983).
[23] E.C. Sherbrooke and N.M. Patrikalakis, Computation of the solutions of nonlinear polynomial systems, Comput. Aided Geom. Design 10 (1993) 379–405.
[24] X.W. Song, T.W. Sederberg, J. Zheng, R.T. Farouki and J. Hass, Linear perturbation methods for topologically consistent representations of free-form surface intersections, Comput. Aided Geom. Design 21 (2004) 303–319.