Proceedings of TMCE 2014, May 19–23, 2014, Budapest, Hungary, edited by I. Horváth and Z. Rusák

AN EMPTINESS TEST AND A STAR TEST FOR PATCHES

Lincong Fang
Le2I Lab, University of Burgundy, 21078 Dijon, France
School of Information Technology,
Zhejiang University of Finance & Economics, 310018 Hangzhou, China
lincong.fang@u-bourgogne.fr

Jean-marc Cane
Dominique Michelucci
Le2I Lab, University of Burgundy, 21078 Dijon, France
{jean-marc.cane, dominique.michelucci}@u-bourgogne.fr

Sebti Foufou
Computer Science and Engineering department, CENG
PO.Box 2713, Qatar University, Qatar
Le2I Lab, University of Burgundy, 21078 Dijon, France
sfoufou@qu.edu.qa

ABSTRACT

A star is a point in a set which can see all of the boundary points. In this paper, algorithms to perform emptiness test and star test for parametric patches are investigated. With those algorithms a studied box is tested whether it is empty, full, or its center is a star point. Otherwise, the tests fail and the box is subdivided. The results are theoretically guaranteed, and will facilitate applying topology analysis algorithms of geometric sets, such as the C.I.A. (Connected components via Interval Analysis) and the H.I.A. (Homotopy type via Interval Analysis). We implemented our algorithms for Bézier patches. Since the interval computation paradigm is used in the tests for implicit sets, the tests for patches are based on the elegant properties of Bézier patches and the de Casteljau algorithm. Our methods may fail because of insufficient accuracy on some particular patches, failure cases are analyzed. Experimental results, including those generated by the C.I.A. algorithm, are given to show the effectiveness of our approach.

KEYWORDS

Geometric modeling, CSG, Bézier patch, critical point, multiple point, star test.

1. INTRODUCTION

Parametric patches such as Bézier patches and spline patches are widely used in computer graphics, computer aided design, and related fields [4]. Bézier patches possess a lot of elegant properties, such as convex hull property, variation diminishing property, affine invariant, which are mathematically convenient. A Bézier patch is defined in terms of tensor product bivariate (in 2D) or trivariate (in 3D) Bernstein polynomials, while all the coefficients construct a control net. The patch can be evaluated using the de Casteljau algorithm [4], which is numerically stable and efficient. For simplicity, we introduce our work for 2D cases.

Bernstein polynomials also have widespread applications in numerical computation, such as curve/surface approximation and interpolation [14], root finding [13], geometric constraint systems solving [7, 12], etc. It is well-known that the de Casteljau algorithm converges much faster than other iterative algorithms. However, the transformation matrix between high-degree canonical power series to Bernstein polynomials is ill-conditioned [5]. Bernstein polytopes [8] are proposed to evaluate or solve high degree and multi-variate systems.

Bézier patches are preferred to implicit sets which are defined by implicit functions, since they are much easier to manipulate, and can construct more free-form shapes. In addition, some common implicit surfaces such as spheres and cylinders can be well approximated or even exactly represented by rational Bézier patches. However, some geometric computations which do not use subdivision are awkward if Bézier shapes are used, for example, it is not so easy to classify an arbitrary point as being inside or outside the shape, while it is trivial for implicit sets. It will be interesting if implicit sets and Bézier shapes can be used to construct geometric models in a uniform way. This is also one of the motivations of this paper.

CSG (constructive solid geometry) is one of the most popular technologies to create a complex surface or object by using Boolean operations. Primitives are the simplest solid objects used for the representation. Usually they are simple shapes: triangles, rectangles, disks, ellipses in 2D, or cuboids, cylinders, prisms, pyramids, spheres, cones in 3D. A convenient property of CSG shapes is that it is easy to do geometric computations like point classification (inside or outside). The computations are dealt for all the underlying primitives and the resulting boolean expression is evaluated. This is a desirable quality for some applications such as collision detection.

One of the most significant features of a geometric shape is its topological properties. Delanoue *et al.* [1, 2] presented an approach to compute topological properties for CSG shapes, whose primitives are all implicit sets. By generating C.I.A. (Connected components via Interval Analysis) and H.I.A. (Homotopy type via Interval Analysis) graphs, a simplicial complex which is homotopy equivalent to the input CSG shape can be obtained. Both C.I.A. and H.I.A. are based on two tests, that is emptiness test and star test. In this paper, approaches of emptiness test and star test for parametric patches are introduced, all the tests are implemented on Bézier patches in 2D.

The rest of the paper is organized as follows. In Section 2, we recall the properties of Bézier patches and basic conceptions introduced by Delanoue [1, 2]. Then methods of emptiness test and star test are introduced in Section 3 and Section 4 respectively. Failure cases are discussed in Section 5 and experimental results are shown in Section 6. At last, we conclude the paper in Section 7.

2. BACKGROUND

2.1. Bézier Curve and Patch

Let $\{\boldsymbol{p}_i = (x_i, y_i), i = 0, ..., n\}$ be the control points, and

$$B_i^n(t) = \binom{n}{i}(1-t)^{n-i}t^i,\ i = 0, ..., n \tag{1}$$

be Bernstein polynomials. A Bézier curve is defined by

$$\boldsymbol{P}(t) = \sum_{i=0}^{n} B_i^n(t)\boldsymbol{p}_i, \quad 0 \le t \le 1 \tag{2}$$

The derivative of the curve has form

$$\boldsymbol{P}'(t) = n\sum_{i=0}^{n-1} B_i^{n-1}(t)\Delta\boldsymbol{p}_i \tag{3}$$

where $\Delta\boldsymbol{p}_i = \boldsymbol{p}_{i+1} - \boldsymbol{p}_i$. The control points construct a control polygon, and there is a well-known VD (variation diminishing) property which may also be generalised into higher dimensions [6].

Property 1. ***(VD Property)*** *If a line is drawn through the curve, the number of intersections with the curve will be less than or equal to the number of intersections with the control polygon.*

Let $\{\boldsymbol{p}_{i,j} = (x_{i,j}, y_{i,j}), i = 0, ..., m, j = 0, ..., n\}$ be the control points, a Bézier patch $\boldsymbol{S}$ is defined by tensor product of Bernstein polynomials

$$\boldsymbol{P}(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n} B_i^m(u)B_j^n(v)\boldsymbol{p}_{i,j},\ 0 \le u, v \le 1 \tag{4}$$

Note that $\boldsymbol{P}(u,v) = (x(u,v), y(u,v))$, and following equation (3), $\boldsymbol{P}'_u(u,v)$ and $\boldsymbol{P}'_v(u,v)$ can also be expressed in Bernstein form

$$\boldsymbol{P}'_u(u,v) = m\sum_{i}^{m-1}\sum_{j}^{n} B_i^{m-1}(u)B_j^n(v)\Delta_i\boldsymbol{p}_{i,j}$$

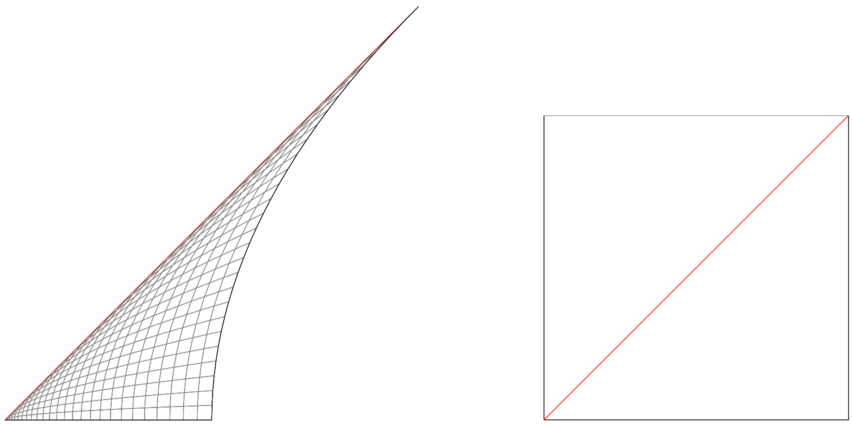
$$\boldsymbol{P}'_v(u,v) = n\sum_{i}^{m}\sum_{j}^{n-1} B_i^{m}(u)B_j^{n-1}(v)\Delta_j\boldsymbol{p}_{i,j}$$

where $\Delta_i\boldsymbol{p}_{i,j} = \boldsymbol{p}_{i+1,j} - \boldsymbol{p}_{i,j}$, $\Delta_j\boldsymbol{p}_{i,j} = \boldsymbol{p}_{i,j+1} - \boldsymbol{p}_{i,j}$. Therefore, Jacobian determinant

$$|J_{\boldsymbol{P}}(u,v)| = \begin{vmatrix} x'_u(u,v) & x'_v(u,v) \\ y'_u(u,v) & y'_v(u,v) \end{vmatrix} \tag{5}$$

can also be expressed in Bernstein form, whose order is $(2m-1) \times (2n-1)$. A point (u_0, v_0) is called a *critical point* if the Jacobian determinant vanishes, i.e., $|J_{\boldsymbol{P}}(u_0, v_0)| = 0$. Moreover, if $x'_u(u_0, v_0) = x'_v(u_0, v_0) = 0$ and $y'_u(u_0, v_0) = y'_v(u_0, v_0) = 0$, the point $\boldsymbol{P}(u_0, v_0)$ is called a *cusp*. If there are two different points in uv-domain, that is $(u_0, v_0) \ne (u_1, v_1)$, such that $\boldsymbol{p} = \boldsymbol{P}(u_0, v_0) = \boldsymbol{P}(u_1, v_1)$, the point $\boldsymbol{p}$ is called a *multiple point*. When there are no multiple points, $\boldsymbol{P} : [0,1]^2 \to \mathbb{R}^2$ is injective, Lagrange et. al [9] proposed a method to perform injectivity test.

Let $\boldsymbol{S}$ be a set in $\mathbb{R}^n$, $\boldsymbol{S}_c$ be the complementary set of $\boldsymbol{S}$, then a point $\boldsymbol{p} \in \boldsymbol{S}$ is called a *boundary point* if and only if $\forall\varepsilon > 0$, $\boldsymbol{N}(\boldsymbol{p}, \varepsilon) \cap \boldsymbol{S}_c \ne \emptyset$, where $\boldsymbol{N}(\boldsymbol{p}, \varepsilon)$ is a ε-neighbourhood of $\boldsymbol{p}$. A set $\boldsymbol{S}$ is called *closed* if $\partial\boldsymbol{S} \subset \boldsymbol{S}$. For parametric patches, the boundary $\partial\boldsymbol{P}([0,1]^2)$ of the patch is a subset of following curves, i.e., four curves $\boldsymbol{P}(\partial[0,1]^2)$, and the critical curve $\boldsymbol{P}(\boldsymbol{K})$ where $\boldsymbol{K}$ is the set of critical points: $\boldsymbol{K} = \{(u,v) \in [0,1]^2 \mid$



(a) (b)

Figure 1 A patch $\boldsymbol{S}$ defined by $\boldsymbol{P}(u,v) = (\frac{1}{2}(u^2+v^2), uv)$, the critical curve is shown in red, both in xy-domain and uv-domain. (a) The contour of the patch. (b) uv-domain.

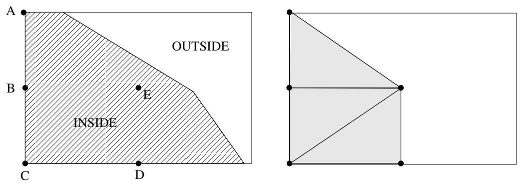


Figure 2 An illustration of the star test in a box.

$|\boldsymbol{P}'|(u,v) = 0\}$. An example is illustrated in Figure 1. Algorithms to trace implicit curves like $\boldsymbol{P}(\boldsymbol{K})$ have been discussed in [10].

2.2. Star test

Here we recall some fundamental concepts and propositions [1, 2].

Definition 1. *A point $\boldsymbol{s}$ is a star for a subset $\boldsymbol{X}$ of an Euclidean set if $\boldsymbol{X}$ contains all the line segments connecting any of its points and $\boldsymbol{s}$.*

Definition 2. *If $\boldsymbol{s}$ is a star for subset $\boldsymbol{X}$ of an Euclidean set, one says that $\boldsymbol{X}$ is star-shaped or $\boldsymbol{s}$-star-shaped.*

Proposition 1. *Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be two $\boldsymbol{s}$-star-shaped sets, then $\boldsymbol{X} \cap \boldsymbol{Y}$ and $\boldsymbol{X} \cup \boldsymbol{Y}$ are also $\boldsymbol{s}$-star-shaped.*

Proposition 2. *Let f be a C^1 function from $\mathbb{R}^n$ to $\mathbb{R}$, $\boldsymbol{B}$ be a convex set and $\boldsymbol{S} = \{\boldsymbol{x} \in \boldsymbol{B} \subset \mathbb{R}^n \mid f(\boldsymbol{x}) \leq 0\}$. If there exists $\boldsymbol{s} \in \boldsymbol{S}$ such that*

$$\{\boldsymbol{x} \in \boldsymbol{B} \mid f(\boldsymbol{x}) = 0,\ \nabla f(\boldsymbol{x}) \cdot (\boldsymbol{x} - \boldsymbol{s}) \leq 0\} = \emptyset$$

then $\boldsymbol{S}$ is $\boldsymbol{s}$-star-shaped.

Figure 2 illustrates the star test in a box through an example: Let $\boldsymbol{S}$ be the patch, and $\boldsymbol{B} = ([x_{min}, x_{max}], [y_{min}, y_{max}])$ be the studied box. Point E is a star for $\boldsymbol{S} \cap \boldsymbol{B}$. Point A is a star for the intersection of $\boldsymbol{S}$ and the upper edge of $\boldsymbol{B}$. Point B is a star for the intersection of $\boldsymbol{S}$ and the left edge of $\boldsymbol{B}$. Point D is a star for the intersection of $\boldsymbol{S}$ and the bottom edge of $\boldsymbol{B}$. The intersection of $\boldsymbol{S}$ and the right edge is empty. Finally, A (and C) are trivial stars for $A \cap \boldsymbol{S}$ (and $C \cap \boldsymbol{S}$). The simplicial complex homotopic to $\boldsymbol{S} \cap \boldsymbol{B}$ is also shown, see Figure 2 right. The maximal simplices are triangles ABE, BCE, CDE. Other simplices of dimension 1 (AB, BE, EA, etc) and dimension 0 (A, B, C, D, E) are deduced by inclusion. By definition of simplical complex, either two simplices intersect on another simplex of the complex, or they are disjoint.

3. EMPTINESS TEST

Generally, a patch may contain critical points and/or multiple points. In this section, we give approaches of emptiness test for a given studied box. Existences of different kinds of points are discussed.

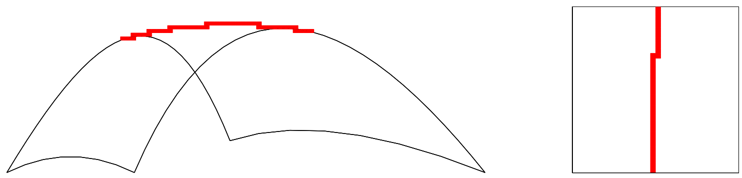
3.1. General points

In this subsection, we present a method to test if a studied box $\boldsymbol{B}$ contains a point of a patch $\boldsymbol{S} = \{\boldsymbol{p} \mid \boldsymbol{p} = \boldsymbol{P}(u,v), 0 \leq u,v \leq 1\}$, which is mainly based on the de Casteljau algorithm.

By the convex hull property, a studied box $\boldsymbol{B}$ is empty if it does not intersect with the bounding box of the control points. As we already know, $\boldsymbol{p}_{0,0} = \boldsymbol{P}(0,0), \boldsymbol{p}_{m,0} = \boldsymbol{P}(1,0), \boldsymbol{p}_{0,n} = \boldsymbol{P}(0,1), \boldsymbol{p}_{m,n} = \boldsymbol{P}(1,1)$, therefore, if one of these four corner points is in the studied box, the box is not empty. Moreover, if $\boldsymbol{B}$ is bounded by four boundary curves $\boldsymbol{P}(\partial[0,1]^2)$, for example if $\forall (x,y) \in \boldsymbol{B}$

$$x_{0,j} \leq x \leq x_{m,j}, j = 0, ..., n$$
$$y_{i,0} \leq y \leq y_{i,n}, i = 0, ..., m \tag{6}$$

we know that $\boldsymbol{B} \subset \boldsymbol{S}$, the box is full as shown in Fig-



(a) (b)

Figure 3 This patch contains A CURVE (in red) of critical points, and multiple points. (a) The contour (including the critical curve in red) of the patch in the xy-domain. (b) The contour of the patch (including the critical curve in red) in the uv-domain.

Algorithm 1 GeneralPointTest($\boldsymbol{B}$, $\boldsymbol{S}$)

Require: $\boldsymbol{B}$ a studied box, $\boldsymbol{S}$ a given patch

Compute bounding box $\boldsymbol{B}_b$ of control points of $\boldsymbol{S}$
if $\boldsymbol{B} \cap \boldsymbol{B}_b = \emptyset$ **then**
  return FALSE
else if $\{\boldsymbol{p}_{00}, \boldsymbol{p}_{m0}, \boldsymbol{p}_{0n}, \boldsymbol{p}_{mn}\} \cap \boldsymbol{B} \neq \emptyset$ **or** $\boldsymbol{B} \subset \boldsymbol{S}$ is guaranteed **then**
  return TRUE
else if $size(\boldsymbol{B}_b) \leq \varepsilon$ **then**
  return UNKNOWN
else
  Subdivide $\boldsymbol{S}$ into $\{\boldsymbol{S}_i\}_{i=0}^{3}$ using the de Casteljau method
  for $i = 0$ to 3 **do**
    $res_i \leftarrow$ GeneralPointTest($\boldsymbol{B}$, $\boldsymbol{S}_i$)
    if res_i = TRUE **then**
      return TRUE
    end if
  end for
  if UNKNOWN $\in \{res_i\}_{i=0}^{3}$ **then**
    return UNKNOWN
  end if
  return FALSE
end if

ure 4. See Algorithm 1 for details.

If the studied box is degenerated to a point (x_0, y_0), it is a test for the existence of the solution in $[0,1]^2$ for such a system of equations

$$x_0 - x(u,v) = 0$$
$$y_0 - y(u,v) = 0$$

There are theorems to guarantee existence of a root of the system inside a box [11].

After the procedure terminates, the box is empty if FALSE is returned. If UNKNOWN is the final result, then it means the studied box may be tangent to the patch on the contour or the intersection is too small, and the accuracy is not sufficient enough to detect it. TRUE and FALSE answers are guaranteed to be correct.

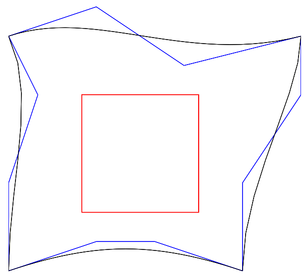


Figure 4 The studied box is bounded by four boundary curves, it is full.

3.2. Critical points

Since $|J_{\boldsymbol{P}}(u,v)|$ can also be expressed in Bernstein form, again the de Casteljau method can be used to test whether a patch contains critical points. Suppose $\{J_{i,j}, i = 0, ..., M, j = 0, ..., N\}$ to be Bernstein coefficients of the Jacobian determinant, then existence of critical points can be guaranteed if

$$max\{J_{0,0}, J_{M,0}, J_{0,N}, J_{M,N}\} \geq 0$$
$$min\{J_{0,0}, J_{M,0}, J_{0,N}, J_{M,N}\} \leq 0$$

Algorithm 2 gives the details of the steps. We return UNKNOWN as the result if the accuracy is not sufficient enough to get a guaranteed result. In Figure 3, we picture the critical curve both in xy-domain and uv-domain.

A point of a patch is a cusp when $x'_u(u,v) = x'_v(u,v) = y'_u(u,v) = y'_v(u,v) = 0$. Since all the items can be expressed in Bernstein form, existence of cusps can be tested using the de Casteljau method. There are cusps if the patch $(x'_u(u,v), x'_v(u,v), y'_u(u,v), y'_v(u,v))$ contains point $(0,0,0,0)$. Or there are no cusps if $(0,0,0,0)$ is out of the patch. Otherwise, the patch needs to be subdivided and the steps are repeated until the bounding box of subdivided patch fulfills a ε condition, in which case UNKNOWN is returned. Since there are more constraints than variables, only non-existence of cusps is proved. In geometric modelling, cusps are not expected.

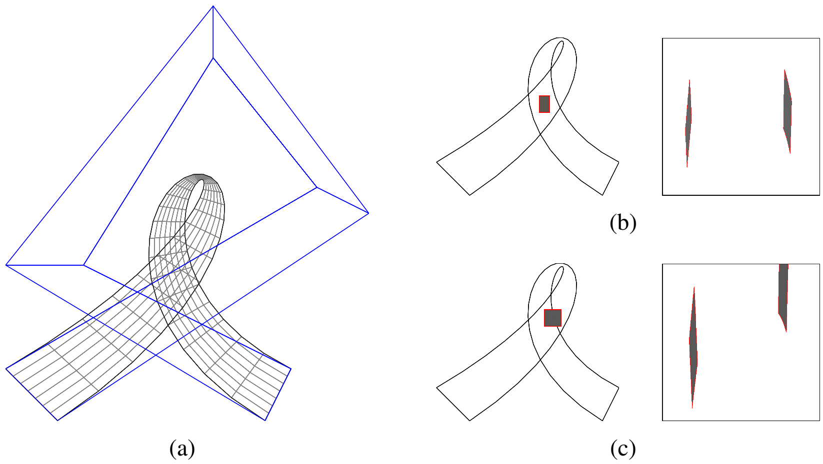


Figure 5 This α shaped patch contains multiple points, but no critical points. (a) The contour of the patch. (b) A studied box which does not intersect with contour curves and its pre-image in uv-domain. (c) A studied box which intersect with one contour curve and its pre-image in uv-domain.

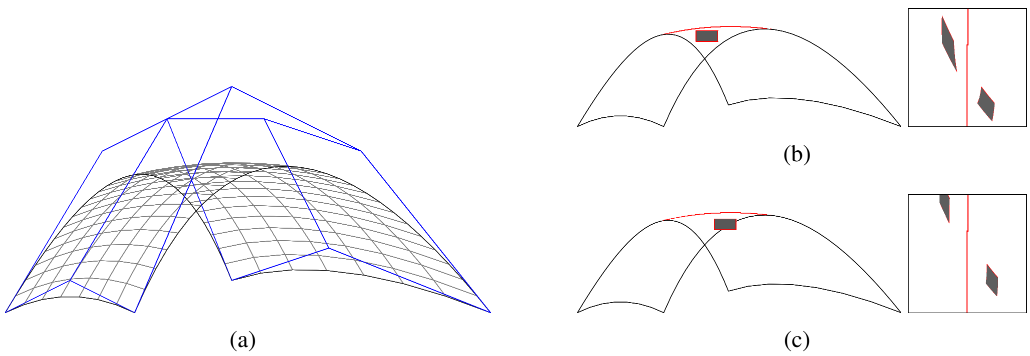


Figure 6 This patch contains a critical curve and multiple points. (a) The contour of the patch. (b) A studied box which does not intersect with contour curves and its pre-image in uv-domain. (c) A studied box which intersect with one contour curve and its pre-image in uv-domain.

Algorithm 2 CriticalPointTest($\boldsymbol{B}$, $\boldsymbol{S}$)

Require: $\boldsymbol{B}$ a studied box, $\boldsymbol{S}$ a given patch
Compute bounding box $\boldsymbol{B}_b$ of control points of $\boldsymbol{S}$
if $\boldsymbol{B} \cap \boldsymbol{B}_b = \emptyset$ **then**
  return FALSE
end if
Compute Jacobian determinant and note $\{J_{i,j}\}$ the Bernstein coefficients
if $max\{J_{i,j}\} < 0$ **or** $min\{J_{i,j}\} > 0$ **then**
  return FALSE
else if $max\{J_{0,0}, J_{M,0}, J_{0,N}, J_{M,N}\} \geq 0$ **and** $min\{J_{0,0}, J_{M,0}, J_{0,N}, J_{M,N}\} \leq 0$ **and** $\boldsymbol{B}_b \subset \boldsymbol{B}$ **then**
  return TRUE
else if $size(\boldsymbol{B}_b) \leq \varepsilon$ **then**
  return UNKNOWN
else
  Subdivide $\boldsymbol{S}$ into $\{\boldsymbol{S}_i\}_{i=0}^{3}$
  for $i = 0$ to 3 **do**
    $res_i \leftarrow$ CriticalPointTest($\boldsymbol{B}$, $\boldsymbol{S}_i$)
    if res_i = TRUE **then**
      return TRUE
    end if
  end for
  if UNKNOWN $\in \{res_i\}_{i=0}^{3}$ **then**
    return UNKNOWN
  end if
  return FALSE
end if

3.3. Multiple points

More generally, a patch may contain multiple points or/and critical points, see Figures 5 and 6. We introduce a sufficient condition that a studied set contains multiple points.

Theorem 1. *Let $\boldsymbol{D} \subset \mathbb{R}^m$ be a closed set, $\boldsymbol{S} = \boldsymbol{P}(\boldsymbol{u})$, $\boldsymbol{u} \in \boldsymbol{D}$ be a parametric set. If for a closed path-connected set $\boldsymbol{B}$, $\boldsymbol{B} \cap \boldsymbol{S} \neq \emptyset$, $\boldsymbol{P}^{-1}(\boldsymbol{B} \cap \boldsymbol{S}) = \cup_i \{\boldsymbol{D}_i\}_{i=0}^{n}$, $\{\boldsymbol{D}_i, i = 0, ..., n\}$ are also closed path-connected sets, and $\boldsymbol{D}_i \cap \boldsymbol{D}_j = \emptyset, i \neq j$, there exist a non-empty set $\boldsymbol{D}_{i_0} \subset \boldsymbol{D} \setminus \partial\boldsymbol{D}$ such that $0 \notin |J_{\boldsymbol{P}}(\boldsymbol{D}_{i_0})|$, then*
1) $\boldsymbol{B} = \boldsymbol{P}(\boldsymbol{D}_{i_0}) \subset \boldsymbol{S} \setminus \partial\boldsymbol{S}$.
2) $\boldsymbol{P} : \boldsymbol{D}_{i_0} \to \boldsymbol{B}$ defined by $\boldsymbol{P}(\boldsymbol{u})$ is injective.
3) $\boldsymbol{B}$ contains multiple points of $\boldsymbol{S}$ if $n > 0$.

Proof. Consider the case when $n = 0$, we have $\boldsymbol{P}(\boldsymbol{D}_0) = \boldsymbol{B} \cap \boldsymbol{S}$. Since $\boldsymbol{D}_{i_0} \subset \boldsymbol{D}$ and $0 \notin |J_{\boldsymbol{P}}(\boldsymbol{D}_{i_0})|$, we know $\boldsymbol{P}(\boldsymbol{D}_0) \cap \partial\boldsymbol{S} = \emptyset$, thus $\boldsymbol{B} \cap \partial\boldsymbol{S} = \emptyset$ and $\boldsymbol{B} \subset \boldsymbol{S} \setminus \partial\boldsymbol{S}$.

Suppose there exist $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ such that $\boldsymbol{P}(\boldsymbol{x}_0) = \boldsymbol{P}(\boldsymbol{x}_1)$, then following the intermediate value theorem, we have $0 \in |J_{\boldsymbol{P}}(\boldsymbol{D}_{i_0})|$ or $\boldsymbol{D}_{i_0} \cap \partial\boldsymbol{D} \neq \emptyset$, which is conflicted with the conditions. Therefore, $\boldsymbol{P} : \boldsymbol{D}_{i_0} \to \boldsymbol{B}$ defined by $\boldsymbol{P}(u, v)$ is injective.

If $n > 0$, since $\boldsymbol{D}_i$ is path-connected, and $\boldsymbol{D}_i \cap \boldsymbol{D}_j = \emptyset, i \neq j$, we can find a closed path-connected set $\widetilde{\boldsymbol{D}}_{i_0} \subset \boldsymbol{D}$ such that $\boldsymbol{D}_{i_0} \subset \widetilde{\boldsymbol{D}}_{i_0}$, $\widetilde{\boldsymbol{D}}_{i_0} \cap \boldsymbol{D}_j = \emptyset, j \neq i_0$. Consider $\widetilde{\boldsymbol{S}} = \boldsymbol{P}(\widetilde{\boldsymbol{D}}_{i_0})$ as a sub-patch, by the case $n = 0$, we know $\boldsymbol{B} \subset \widetilde{\boldsymbol{S}} \subset \boldsymbol{S}$. Furthermore, $\forall \boldsymbol{p} \in \boldsymbol{D}_i, i \neq i_0$, $\boldsymbol{p}$ is a multiple point. □

Algorithm 3 preImage($\boldsymbol{B}, \boldsymbol{S}, \boldsymbol{node}$)

Require: $\boldsymbol{B}$ a studied box, $\boldsymbol{S}$ a given patch, $\boldsymbol{node}$ a quad tree node whose data is the parametric domain of $\boldsymbol{S}$
Compute bounding box $\boldsymbol{B}_b$ of control points of $\boldsymbol{S}$
if $\boldsymbol{B}_b \subset \boldsymbol{B}$ **then**
  Label $\boldsymbol{node}$ INSIDE
else if $\boldsymbol{B}_b \cap \boldsymbol{B} = \emptyset$ **then**
  Label $\boldsymbol{node}$ OUTSIDE
else if $size(\boldsymbol{B}_b) < \varepsilon$ **then**
  Label $\boldsymbol{node}$ BOUNDARY
else
  Subdivide $(\boldsymbol{S}, \boldsymbol{node})$ into $\{(\boldsymbol{S}_i, \boldsymbol{n}_i)\}_{i=0}^{3}$
  for $i = 0$ to 3 **do**
    preImage($\boldsymbol{B}$, $\boldsymbol{S}_i$, $\boldsymbol{n}_i$)
  end for
end if

Now we give a method to test the existence of multiple points. Let $\boldsymbol{U}_0 = [0, 1]^2$, Algorithm 3 constructs a quad tree in uv-domain. Actually, in Algorithm 3, the de Casteljau method on $(x, y, u, v) \in (\boldsymbol{S}, \boldsymbol{U}_0)$ is used. By the adjacency of boxes in the tree, all boxes which are not OUTSIDE can be clustered into different components. Let $\{\widetilde{\boldsymbol{D}}_i, i = 0, ..., n\}$ be these components, then we have

$$\boldsymbol{P}^{-1}(\boldsymbol{B} \cap \boldsymbol{S}) \subset \bigcup_{i=0}^{n} \{\widetilde{\boldsymbol{D}}_i\}$$

From Theorem 1, if there is a $\widetilde{\boldsymbol{D}}_i \in (0, 1)^2$ which contains no critical points, then $\boldsymbol{B} \subset \boldsymbol{S}$ is guaranteed, also $\boldsymbol{B}$ contains multiple points if $n > 0$. Otherwise, the studied box $\boldsymbol{B}$ should be subdivided and the test is performed for each small box. Once a ε condition in xy-domain is fulfilled and no guaranteed result can be obtained, UNKNOWN is returned.

In Figure 5, an α shaped patch is given, two different studied boxes included by the patch and their preimages are shown in Figures 5(b) and 5(c). For Figure 5(b), the pre-image of the studied box contains two path-connected components, none of which are intersected with $\partial[0,1]^2$, thus the studied box is proved to be in the patch. The studied box in Figure 5(c) is intersected with one of the four curves $\boldsymbol{P}(\partial[0,1]^2)$, and its pre-image also contains two path-connected components. One of them is intersected with $\partial[0,1]^2$, while the other is not, so the box is also in the patch.

Figure 6 gives another example, where the given patch contains critical points and multiple points. None of the two components of the pre-image of the studied box in Figure 6(b) are intersected with $\partial[0,1]^2$, or contain critical points, thus the studied box is guaranteed to be in the patch. In Figure 6(c), the studied box is intersected with one of the contour curves, and its pre-image contains two components. One of them is not intersected with $\partial[0,1]^2$, and contains no critical points, so the box is in the patch.

4. STAR TEST

C.I.A. and H.I.A. need to test if a given box $\boldsymbol{B}$ in the visible space (x,y) is empty ($\boldsymbol{B} \cap \boldsymbol{P}([0,1]^2) = \emptyset$), or full ($\boldsymbol{B} = \boldsymbol{B} \cap \boldsymbol{P}([0,1]^2)$), or, when $\boldsymbol{B}$ contains boundary points, if the center of $\boldsymbol{B}$ is a star. If the star test fails then the box $\boldsymbol{B}$ is subdivided. In this section, only the case $\boldsymbol{B}$ contains boundary points is considered, other cases can be tested using methods in previous section.

By the definition, we know that $\boldsymbol{S}$ is $\boldsymbol{s}$-star-shaped if $\forall \boldsymbol{p} \in \partial\boldsymbol{S}, \{\boldsymbol{p}\} = \{(1-t)\cdot\boldsymbol{s}+t\cdot\boldsymbol{p} \mid t > 0\} \cap \partial\boldsymbol{S}$, which is a test for the uniqueness of a solution [15, 16, 3]. Since $\partial\boldsymbol{S} \subset \boldsymbol{P}(\partial[0,1]^2) \cup \boldsymbol{P}(\boldsymbol{K})$, and $\boldsymbol{P}(\partial[0,1]^2)$ are Bézier curves, the uniqueness of intersection of a line and $\boldsymbol{P}(\partial[0,1]^2)$ can be tested according the variation diminishing property. In other words, if a parametric patch $\boldsymbol{S}$ without critical points is bounded by a Bézier curve, then a point $\boldsymbol{s} \in \boldsymbol{S}$ is a star of $\boldsymbol{S}$ if it is a star of the control polygons. This gives a basic idea for the star test of regular patches.

However, a general patch may contain multiple and/or critical points, which makes the test more difficult. We recall a sufficient condition for the uniqueness of intersection of two curves [15, 16]. Let $\boldsymbol{P}(t)$ be a Bézier curve, and $\{\boldsymbol{p}_i, i = 0,...,n\}$ be the control points, $\boldsymbol{T}$ denoted the set of positive linear combinations of $\Delta\boldsymbol{p}_i = \boldsymbol{p}_{i+1} - \boldsymbol{p}_i$, which is called *tangent cone* [15]. Two curves can not intersect more than once if their tangent cones are not overlapping [15, 16].

Let $\partial\boldsymbol{S} = \cup_{i=0}^{n}\{\boldsymbol{S}'_i\} \cup \boldsymbol{P}(\boldsymbol{K})$ be a partition of the boundary of a patch, for $\boldsymbol{s} \in \boldsymbol{S}$, let $\boldsymbol{S}_i = \{(1-t)\cdot\boldsymbol{s}+t\cdot\boldsymbol{p} \mid 0 \le t \le 1, \boldsymbol{p} \in \boldsymbol{S}'_i\}$ and $\boldsymbol{S}_{\boldsymbol{K}} = \{(1-t)\cdot\boldsymbol{s}+t\cdot\boldsymbol{p} \mid 0 \le t \le 1, \boldsymbol{p} \in \boldsymbol{P}(\boldsymbol{K})\}$, $\boldsymbol{T}_i$ be the tangent cone of $\boldsymbol{S}'_i$. Then there are no loops composed by curves $\boldsymbol{P}(\partial[0,1]^2)$ if $\boldsymbol{T}_i \cap \boldsymbol{T}_j = \emptyset$ where $i \ne j$ and $\partial\boldsymbol{S}_i \subsetneq \boldsymbol{S}_i \cap \boldsymbol{S}_j$ [15].

In this paper, we assume there are no loops composed by critical curves for a given patch. Thus, $\boldsymbol{S} = \cup_i^n\{\boldsymbol{S}_i\} \cup \boldsymbol{S}_{\boldsymbol{K}}$ if $\{(1-t)\cdot\boldsymbol{s}+t\cdot\boldsymbol{p} \mid \boldsymbol{p} \in \partial\boldsymbol{S}', 0 \le t \le 1\} \subset \boldsymbol{S}$ and $\{(1-t)\cdot\boldsymbol{s}+t\cdot\boldsymbol{p} \mid \boldsymbol{p} \in \partial\boldsymbol{S}_{\boldsymbol{K}}, 0 \le t \le 1\} \subset \boldsymbol{S}$. Furthermore, $\boldsymbol{s}$ is a star of $\boldsymbol{S}$ if it is a star of each $\boldsymbol{S}_i$ and $\boldsymbol{S}_{\boldsymbol{K}}$.

We give a method to do the star test for patch containing no critical points, details are illustrated as the Algorithm 4 and 5. Algorithm 4 gives a star test for regular patches, which contain no multiple points and no critical points. Algorithm 5 needs to be run for the star test of patches with multiple points but no critical points.

Patches without cusps are considered in this paper, so $\boldsymbol{P}'(u,v)$ is continuous and there is no more than one critical point on a line connecting $\boldsymbol{s} \in \boldsymbol{S}$ and a point on the critical curve if $\{(\boldsymbol{P}(u,v) - \boldsymbol{s}) \cdot \nabla\boldsymbol{P}(u,v) = 0\} = \emptyset$, which can be tested using interval arithmetic method. We also considered to use the de Casteljau method, but instability happens because of high degree of the equation. For patches containing critical points, another test whose steps are similar to Algorithm 4 shall be run to complete the star test. The differences are the stack $\mathcal{L}_{stack}$ saves the patches containing critical points, and interval arithmetic method is used for the star test of $\boldsymbol{S}_{\boldsymbol{K}}$.

The star test fails if FALSE or UNKNOWN is returned, different cases of failure are discussed in Section 5.

5. LIMITATIONS

Delanoue's method fails when the studied set is not fat (a set is fat when it is equal to the closure of its interior), or when the boundary is tangent to one of the subdivision hyperplane (a line in 2D, a plane in 3D): in this latter case, no point can be found (thus no star point can be found) inside the intersection of the studied object and of the subdivision hyperplane: thus Delanoue's method fails. Our test also fails in the same cases. To discard the first case of failure, Delanoue assumes the studied set is fat; we also assume that the studied set is fat (otherwise it can not be manufactured). He used a probabilistic argument to discard the second case of failure: each subdivision hyperplane in the visible space x,y or x,y,z is randomly perturbed, and the probability for an hyperplane to be tangent to an object is zero. If a failure is due to this kind of accident, re-running the program will succeed with proba-

Algorithm 4 RegularStarTest($\mathcal{S}$, $\boldsymbol{B}$, $\boldsymbol{S}$)

Require: $\boldsymbol{B}$ a studied box, $\boldsymbol{S}$ a given patch, $\boldsymbol{B} \cap \boldsymbol{S} \neq \emptyset$

Initialization: $\mathcal{S} = \emptyset$
for all $\boldsymbol{P}(t) \in \boldsymbol{P}(\partial[0,1]^2)$ **do**
  $t_{min} = \min\{t | \boldsymbol{P}(t) \in \boldsymbol{B}\}$
  $t_{max} = \max\{t | \boldsymbol{P}(t) \in \boldsymbol{B}\}$
  Stack $\boldsymbol{P}(t), t_{min} - \varepsilon \leq t \leq t_{max} + \varepsilon$ in $\mathcal{L}_{statck}$
end for
$\boldsymbol{s} = center(\boldsymbol{B})$
if $\boldsymbol{s} \notin \boldsymbol{S}$ **then**
  return FALSE
end if
while $\mathcal{L}_{statck} \neq \emptyset$ **do**
  Pop $\mathcal{L}_{statck}$ into $\boldsymbol{C}$
  Compute the bounding box $\boldsymbol{B}_b$ of $\boldsymbol{C}$
  if not ($\boldsymbol{B} \cap \boldsymbol{B}_b = \emptyset$ **or** $\boldsymbol{B}_b \subset \boldsymbol{S}$) **then**
    if not $\{(1-t) \cdot \boldsymbol{s} + t \cdot \boldsymbol{p} | 0 \leq t \leq 1, \boldsymbol{p} \in \partial \boldsymbol{C}\} \subset \boldsymbol{S}$ **then**
      return FALSE
    else if $\{(1-t) \cdot \boldsymbol{s} + t \cdot \boldsymbol{p} | 0 \leq t \leq 1, \boldsymbol{p} \in \boldsymbol{C}\}$ is $\boldsymbol{s}$-star-shaped **then**
      Push $\boldsymbol{C}$ in $\mathcal{S}$
    else if $size(\boldsymbol{B}_b) < \varepsilon$ **then**
      return UNKNOWN
    else
      Subdivide $\boldsymbol{C}$ into $\{\boldsymbol{C}_i\}$
      for all $\boldsymbol{C}_i = \boldsymbol{P}(t)$ **do**
        $t_{min} = \min\{t | \boldsymbol{P}(t) \in \boldsymbol{B}\}$
        $t_{max} = \max\{t | \boldsymbol{P}(t) \in \boldsymbol{B}\}$
        Stack $\boldsymbol{P}(t), t_{min} - \varepsilon \leq t \leq t_{max} + \varepsilon$ in $\mathcal{L}_{statck}$
      end for
    end if
  end if
end while
return TRUE

Algorithm 5 CrossStarTest($\mathcal{S}$, $\boldsymbol{B}$, $\boldsymbol{S}$)

Require: $\mathcal{S}$ result of Algorithm 4, $\boldsymbol{B}$ a studied box, $\boldsymbol{S}$ a given patch

while $\mathcal{S} \neq \emptyset$ **do**
  Pop $\mathcal{S}$ into $\boldsymbol{C}$
  $\overline{\boldsymbol{C}} = \{(1-t) \cdot \boldsymbol{s} + t \cdot \boldsymbol{p} | 0 \leq t \leq 1, \boldsymbol{p} \in \boldsymbol{C}\}$
  Compute the tangent cone $\boldsymbol{T_C}$ of $\boldsymbol{C}$
  for all $\boldsymbol{C}_i \in \mathcal{S}$ **do**
    $\overline{\boldsymbol{C}_i} = \{(1-t) \cdot \boldsymbol{s} + t \cdot \boldsymbol{p} | 0 \leq t \leq 1, \boldsymbol{p} \in \boldsymbol{C}_i\}$
    Compute the tangent cone $\boldsymbol{T}_i$ of $\boldsymbol{C}_i$
    if $\partial \overline{\boldsymbol{C}} \subsetneq \overline{\boldsymbol{C}} \cap \overline{\boldsymbol{C}_i}$ **and** $\boldsymbol{T_C} \cap \boldsymbol{T}_i \neq \emptyset$ **then**
      if $\boldsymbol{T_C} > \boldsymbol{T}_i$ **then**
        Subdivide $\boldsymbol{C}$ into $\{\boldsymbol{C}_j\}$
      else
        Subdivide $\boldsymbol{C}_i$ into $\{\boldsymbol{C}_j\}$
        Remove $\boldsymbol{C}_i$ from $\mathcal{S}$
      end if
      for all $\{\boldsymbol{C}_j = \boldsymbol{P}(t)\}$ **do**
        if not $\{(1-t) \cdot \boldsymbol{s} + t \cdot \boldsymbol{p} | 0 \leq t \leq 1, \boldsymbol{p} \in \partial \boldsymbol{C}_j\} \subset \boldsymbol{S}$ **then**
          return FALSE
        else if $\boldsymbol{C}_j \cap \boldsymbol{B} \neq \emptyset$ **then**
          $t_{min} = \min\{t | \boldsymbol{P}(t) \in \boldsymbol{B}\}$
          $t_{max} = \max\{t | \boldsymbol{P}(t) \in \boldsymbol{B}\}$
          Push $\boldsymbol{P}(t), t_{min} - \varepsilon \leq t \leq t_{max} + \varepsilon$ into $\mathcal{S}$
        end if
      end for
      break
    end if
  end for
end while
return TRUE

bility 1.

Actually, Delanoue's method may fail in a third case: when some parts of the studied object are so thin that the accuracy of the floating point arithmetic (which is used by the interval arithmetic) is insufficient, or all the memory of the computer is used.

There are several particular cases of patches which may cause failure, first when there are cusps on the patch. For a cusp, where the patch is degenerate, none zero normal vector can be computed and the star test will fail. A second failure case is that the boundary curves of the patch are tangent or nearly tangent to each other such that a ring is formed, see Figure 7. Our method can not detect the coincidence of the boundary curves, and it will fail the test. A first solution is to manage adjacency relations, like the BReps; a second solutions is to make the patch overlap, or to split the patch in two overlapping patches. When one of these cases occur, we return an error code.

6. EXPERIMENTAL RESULTS

In our implementation, if the star test of a studied box returns FAIL or UNKNOWN, we subdivide the studied box along the longest edge at a random position. Furthermore, if a subdivided box fulfills a ε condition, we halt the subdivided procedure and restart the test. A maximum number for repeating is set to halt the whole procedure. In this case, we treat it as a case of failure and return an error code.

We have applied the above tests to generate the path-connected components for various geometric patches

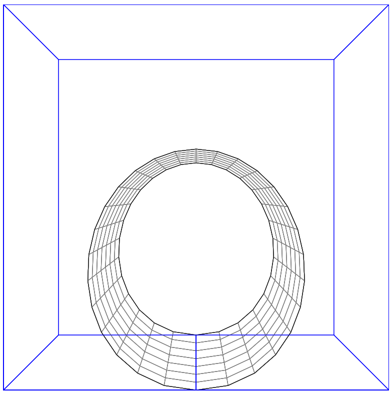


Figure 7 A patch forms a ring when two boundary curves coincide.

using our methods. The results are illustrated in Figures 8, 9, and 10. In all these examples, we show the contour of the patch, the uv-domain, the C.I.A. decompositions using the proposed methods. Figure 8 shows a Bézier patch representing the number 6 whose control polygon is not self-intersecting, this patch contains multiple points, but no critical points, it is not regular. Figures 8(c) and 8(d) give two different results of the star test after 232 and 896 trials (calls to the star test procedure), the studied box covers only part of the patch where it is self-overlapping. Figure 9 shows a Bézier patch whose control polygon is self-intersecting, Figures 9(c) and 9(d) give two different results of the star test after 552 and 945 trials. Figure 10 shows a Bézier patch which contains a critical curve and multiple points, it is not a regular patch. Figures 10(c) and 10(d) give two different results of the star test after 63 and 75 trials. Example such as the patch in Figure 1 fails the star test, since the point $\boldsymbol{P}(1,1)$ can only be seen from the points on the critical curve.

In Figure 11, we give some examples of the star test of CSG shapes containing Bézier patch as one of the primitives. Figure 11(c) gives a CSG shape which is the union of a patch and a disk, and the star test after 265 trials. Figure 11(d) gives a CSG shape which is the difference of a patch and a disk, and the star test after 4690 trials.

7. CONCLUSION

The emptiness test and the star test are fundamental for the C.I.A. and the H.I.A. algorithms. In this paper, we studied the tests for parametric patches. A parametric patch is the image of a function defined on the parametric domain; however, the boundary of the patch does not have to be the image of the boundary of the domain which makes the tests non-trivial. To perform the tests for Bézier patches, we use the convex hull property, the variation diminishing property, and the de Casteljau algorithm. A general parametric patch does not possess such elegant properties, thus we will have to investigate other new ways to implement the tests.

Moreover, several other challenging problems remain to be studied. The star test of the complement set for a given patch is necessary for a general CSG shape, which is non-trivial when there are multiple points and/or critical points. Implementation of the tests for spline patches, and other more general parametric patches shall be considered. Also, it will be of great value to extend the tests to 3D shapes. Following all these results, more complicated geometric sets will be taken into consideration, such as shapes formed by sweeping, Minkowski sum, etc.

ACKNOWLEDGEMENTS

This publication was made possible by NPRP grant #09-906-1-137 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

[1] Nicolas Delanoue, Luc Jaulin, and Bertrand Cottenceau. Using interval arithmetic to prove that a set is path-connected. *Theor. Comput. Sci.*, 351(1):119–128, 2006.

[2] Nicolas Delanoue, Luc Jaulin, and Bertrand Cottenceau. Guaranteeing the homotopy type of a set defined by non-linear inequalities. *Journal of Reliable Computing*, 13(5):381–398, 2007.

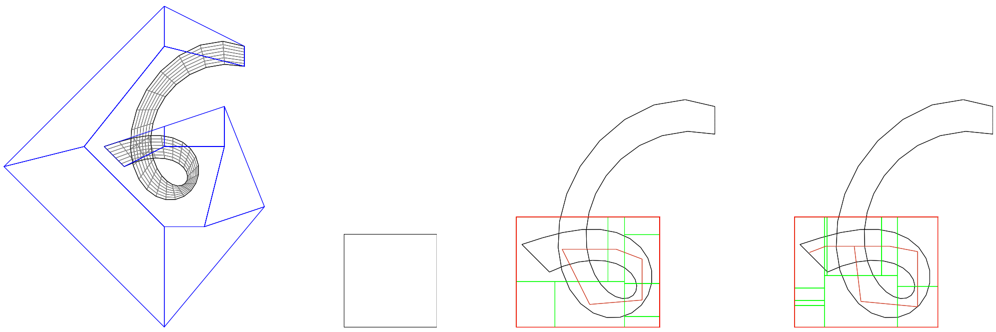
[3] Gershon Elber and Myung-Soo Kim. Geometric constraint solver using multivariate rational spline functions. In *Proceedings of the sixth ACM symposium on Solid modeling and applications*, pages 1–10, New York, NY, USA, 2001. ACM.

[4] Gerald E. Farin. *Curves and Surfaces for CAGD: a practical guide*. Morgan Kaufmann, 2002.

[5] Rida T. Farouki. On the stability of transformations between power and Bernstein polynomial forms. *Computer Aided Geometric Design*, 8(1):29–36, 1991.

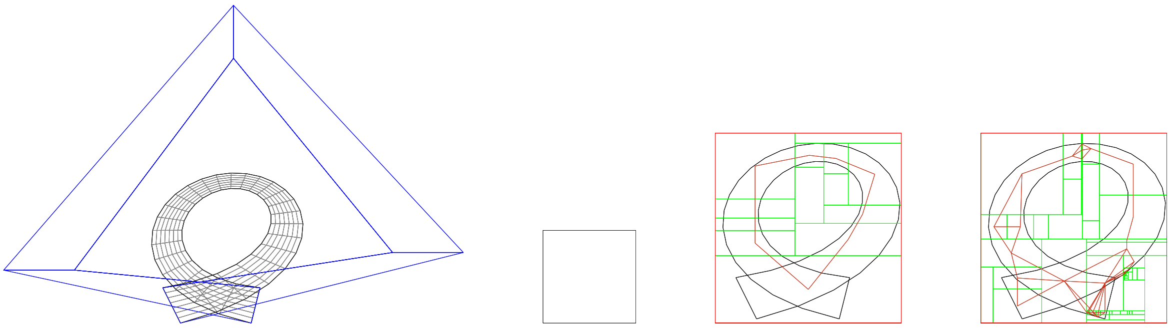
[6] Rida T. Farouki. *Pythagorean-Hodograph Curves: Algebra and Geometry Inseparable*. Springer, 2007.

[7] Sebti Foufou and Dominique Michelucci. Bernstein basis and its application in solving geomet-



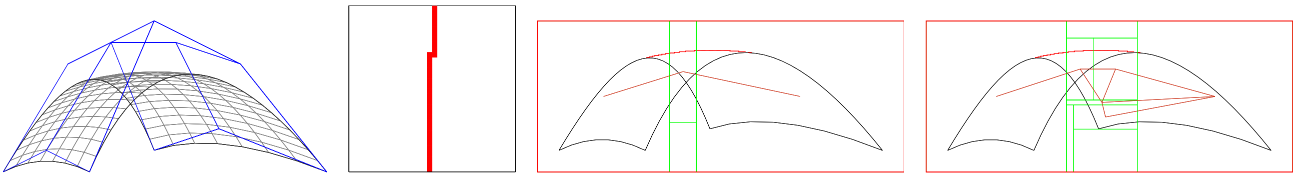
(a) (b) (c) (d)

Figure 8 This 6 shaped Bézier patch contains multiple points, but no critical points. It is not regular. (a) The contour of the patch. (b) uv-domain. (c) The star test after 232 trials and C.I.A. generated using our method. (d) The star test after 896 trials and C.I.A. generated using our method.



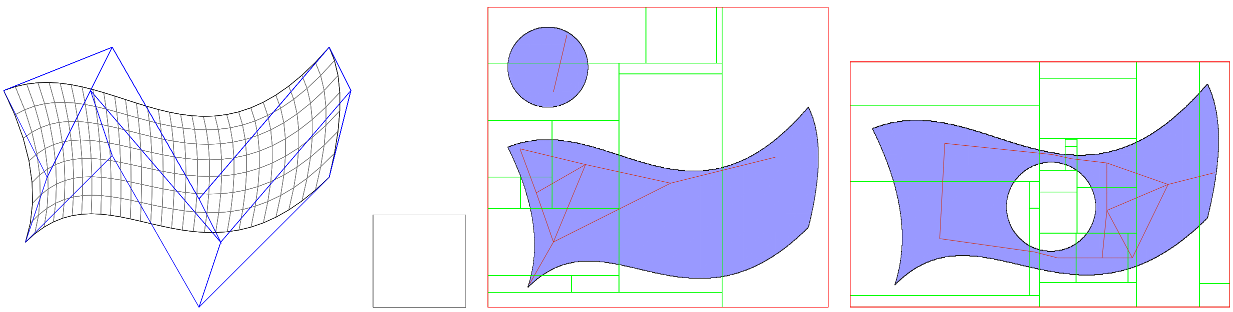
(a) (b) (c) (d)

Figure 9 This α shaped Bézier patch contains multiple points, but no critical points. It is not regular. (a) The contour of the patch. (b) uv-domain. (c) The star test after 552 trials and C.I.A. generated using our method. (d) The star test after 945 trials and C.I.A. generated using our method.



(a) (b) (c) (d)

Figure 10 This Bézier patch contains a critical curve and multiple points. It is not regular. (a) The contour of the patch. (b) uv-domain. (c) The star test after 63 trials and C.I.A. generated using our method. (d) The star test after 75 trials and C.I.A. generated using our method.



(a) (b) (c) (d)

Figure 11 Examples of the star test of CSG shapes. (a) This Bézier patch is regular. (b) uv-domain. (c) The union of a patch and a disk, and the star test after 265 trials and C.I.A. generated using our method. (d) The difference of a patch and a disk, and the star test after 4690 trials and C.I.A. generated using our method.

ric constraint systems. *Journal of Reliable Computing*, 17:192–208, 2012.

[8] Christoph Fünfzig, Dominique Michelucci, and Sebti Foufou. Polytope-based computation of polynomial ranges. *Computer Aided Geometric Design*, 29(1):18–29, 2012.

[9] Sebastien Lagrange, Nicolas Delanoue, and Luc Jaulin. On sufficient conditions of the injectivity: Development of a numerical test algorithm via interval analysis. *Journal of Reliable Computing*, 13(5):409–421, 2007.

[10] Ralph Martin, Huahao Shou, Irina Voiculescu, Adrian Bowyer, and Guojin Wang. Comparison of interval methods for plotting algebraic curves. *Computer Aided Geometric Design*, 19(7):553 – 587, 2002.

[11] J.-P. Merlet. ALIAS: an interval analysis based library for solving and analyzing system of equations. In *Systèmes d'Équations Algébriques*, Toulouse, France, 2000.

[12] Dominique Michelucci, Sebti Foufou, and Arnaud Kubicki. On the complexity of the Bernstein combinatorial problem. *Journal of Reliable Computing*, 17:22–33, 2012.

[13] Bernard Mourrain and Jean P. Pavone. Subdivision methods for solving polynomial equations. *Journal of Symbolic Computation*, 44(3):292–306, 2005.

[14] Thomas W. Sederberg and Rida T. Farouki. Approximation by interval Bézier curves. *Computer Graphics and Applications*, 12(5):87–95, 1992.

[15] Thomas W. Sederberg and Ray J. Meyers. Loop detection in surface patch intersections. *Computer Aided Geometric Design*, 5(2):161 – 171, 1988.

[16] Thomas W. Sederberg and Alan K. Zundel. Pyramids that bound surface patches. *Graphical Models and Image Processing*, 58(1):75 – 81, 1996.