# ON LOCAL INJECTIVITY OF 2D TRIANGULAR CUBIC BEZIER FUNCTIONS

Victoria Hernández-Mederos , Jorge Estrada-Sarlabous * Dionne León Madrigal *
* Instituto de Cibernética, Matemática y Física, La Habana, Cuba
e-mails: vicky@icmf.inf.cu, jestrada@icmf.inf.cu, dionne@icmf.inf.cu

**ABSTRACT**
In this paper we obtain a necessary and sufficient condition for local injectivity of a 2D triangular cubic Bezier function. The condition can be easily checked since it reduces the analysis of the local injectivity to determine if a quartic plane algebraic curve cuts a triangle. An original algorithm to test if a plane algebraic curve of any degree passes through a triangle was developed to verify the previous condition. The algorithm is based on subdivision and range analysis for a triangular region. Additionally, we obtain a sufficient condition for local injectivity of a 2D triangular cubic Bezier function.

**RESUMEN**
En este artículo presentamos una condición necesaria y suficiente para la inyectividad local de funciones triangulares de Bezier en 2D. En este caso el análisis de la inyectividad local se reduce a determinar si una curva algebraica plana de grado 4 pasa por un triángulo. Para verificar la condición anterior, presentamos un algoritmo novedoso para detectar si una curva algebraica plana de grado arbitrario pasa por un triángulo. El algoritmo utiliza técnicas de subdivisión y análisis de rango de una función. Además, presentamos una condición suficiente para la inyectividad local de una función triangular de Bezier en 2D.

**Key words:** injectivity, triangular Bezier functions

**MSC:** 65D17, 65D07

## 1. INTRODUCTION.

Mapping functions transforming certain regions into themselves are very important in Computer Graphics and related areas. The injectivity of such functions is crucial in image warping and morphing [Wolberg, 1990], free form deformation [Sedenberg-Parry, 1986] and surface parameterization [Floater-Horman, 2004]. In image morphing the non injectivity may introduce wrinkles in the resulting image since some areas of the original image are folded. In surface parameterization, local injectivity of mapping functions is the basis for avoiding self-intersections on regular surfaces.

Some approaches to the local injectivity issue have been already presented in the literature. In [Floater-Gostman, 1999] it is described a method for morphing tilings injectively based on convex combinations. Several methods are known to generate injective mapping functions for image warping and morphing see [Fujimura-Makarov, 1998], [Lee et. al., 1995], [Lee et. al., 1996], [Lee et. al., 1996a)]. In [Goodman-Unsworth, 1996] a sufficient condition for the injectivity of a 2D Bezier surface is presented. This condition can be also applied to a 2D B-spline function. Since 2D and 3D B-spline functions have been adopted by some authors to generate mapping functions, sufficient conditions for the local injectivity of cubic 2D and 3D B-spline functions were obtained in [Choi-Lee, 1999].

In the case of triangulated domains the simplest mapping function is a piecewise linear map. A special class of these maps are convex combination mappings introduced by Floater in [Floater, 1997]. In [Floater, 2003] a sufficient condition for the injectivity of convex combination mappings over triangulations can be found. Since piecewise linear maps are only $C^0$ continuous, if higher continuity order is required it is natural to use triangular Bezier spline surfaces to define mapping functions. To the best of our knowledge, not results have been reported concerning the injectivity of this class of mappings.

In this paper we obtain a sufficient condition for local injectivity of a 2D triangular cubic Bezier function similar to the one presented in [Choi-Lee, 1999] for uniform cubic B-spline functions. Additionally, we present a necessary and sufficient condition, which reduces the analysis of the local injectivity to determine if a quartic plane algebraic curve cuts the canonical triangle. To verify this condition we develop an original algorithm to check if a plane algebraic curve of any degree passes through a triangle. The algorithm is based on subdivision and range analysis for a triangular region.

## 2. PRELIMINARIES.

Let $f : R^2 \to R^2$ be a smooth spline function, consisting of cubic triangular Bezier patches, each one determined by 10 control points in $R^2$. The injectivity of the function $f$ may fail in two cases. First, a global violation may occur if the control net contains self-intersections. Due to the convex hull property of Bernstein polynomials, if control net has not self-intersections, only a local violation of the injectivity may appear. Since in most of the applications it is possible to avoid self-intersecting control nets, in this paper we only investigate the local injectivity of triangular cubic Bezier spline functions. But, if we are only interested on local injectivity, then it is sufficient to focus on a single patch of $f$, since the same conditions can be applied to the rest of the patches. In consequence, in the rest of this paper we denote by $f$ a single patch of a 2D triangular cubic Bezier spline function defined on a triangular domain $T$.

The function $f$ can be written in barycentric coordinates $u, v \geq 0, \quad u + v \leq 1$ with respect to the vertices of the triangle $T$ as $f(u,v) = (x(u,v), y(u,v))$ where $x(u,v)$ and $y(u,v)$ are given by,

$$x(u,v) = \sum_{i+j+k=3} b^1_{ijk} B^3_{ijk}(u,v) \tag{1}$$

$$y(u,v) = \sum_{i+j+k=3} b^2_{ijk} B^3_{ijk}(u,v) \tag{2}$$

$B^3_{ijk}(u,v)$ are bivariate Bernstein polynomials of degree 3,

$$B^3_{ijk}(u,v) = \frac{3!}{i!\,j!\,k!} u^i v^j (1-u-v)^{3-i-j}, \quad i + j + k = 3 \tag{3}$$

and $b^1_{ijk}, b^2_{ijk}$ with $i + j + k = 3$ are real numbers.

Observe that the map assigning to a point $(x, y) \in T$ its barycentric coordinates *(u,v,1-u-v)*, with respect to the vertices of $T$, is a $C^\infty$-diffeomorphism. Therefore, we may translate the proof of the local injectivity of the map $f$ defined on $T$, to the proof of the local injectivity of the map induced by $f$ from the canonical triangle $\tilde{T}$ with vertices $(0,0), (1,0), (0,1)$ to $R^2$. For the sake of simplicity we also denote that induced map by $f$.

It is well known that a function $f : R^2 \to R^2$ is locally injective if and only if its Jacobian matrix is nonsingular over its domain [Buck, 1978]. The Jacobian matrix of $f$ is given by,

$$Jf = \begin{bmatrix} x_u & x_v \\ y_u & y_v \end{bmatrix}$$

## 3. LOCAL INJECTIVITY OF 2D TRIANGULAR CUBIC BEZIER FUNCTIONS.

### 3.1 Sufficient condition

In this section we use the ideas contained in [Choi-Lee, 1999] to obtain a sufficient condition for local injectivity of a 2D triangular Bezier function. The injectivity of function $f$ depends on the configuration of the 10 control points

$$b_{ijk} = (b^1_{ijk}, b^2_{ijk}), \quad i + j + k = 3.$$

It is well known [Farin, 1993] that bivariate Bernstein polynomials of any degree *n* have linear precision, i.e,

$$u = \sum_{i+j+k=n} \frac{i}{n} B^n_{ijk}(u,v)$$
$$v = \sum_{i+j+k=n} \frac{j}{n} B^n_{ijk}(u,v)$$

Thus, if $b_{ijk} = b^0_{ijk} := (i/3, j/3)$, then $f(u,v)$ is the identity function. In consequence, a general 2D triangular Bezier function $f(u,v) = (x(u,v), y(u,v))$ may be written as,

$$f(u,v) = (u,v) + \sum_{i+j+k=3} \Delta b_{ijk} B^3_{ijk}(u,v) \tag{4}$$

where $\Delta b_{ijk} = b_{ijk} - b^0_{ijk} := (\Delta x_{ijk}, \Delta y_{ijk}) = (b^1_{ijk} - i/3, b^2_{ijk} - j/3)$. Therefore

,

$$x_u = 1 + \sum_{i+j+k=3} \Delta x_{ijk} \frac{\partial}{\partial u} B^3_{ijk}(u,v)$$
$$x_v = \sum_{i+j+k=3} \Delta x_{ijk} \frac{\partial}{\partial v} B^3_{ijk}(u,v)$$
$$y_u = \sum_{i+j+k=3} \Delta y_{ijk} \frac{\partial}{\partial u} B^3_{ijk}(u,v)$$
$$y_v = 1 + \sum_{i+j+k=3} \Delta y_{ijk} \frac{\partial}{\partial v} B^3_{ijk}(u,v)$$

Denote by $r_1 = (x_u, x_v)$ and $r_2 = (y_u, y_v)$ the row vectors of the Jacobian matrix *Jf*. Observe that $r_1$ and $r_2$ are functions from $\tilde{T}$ to $R^2$ that depend on *u* and *v*. With this notation *f* is locally injective on $\tilde{T}$ if and only if vectors $r_1$ and $r_2$ are linearly independent for all $(u,v) \in \tilde{T}$.

As Choi and Lee noticed in [Choi-Lee, 1999] associating to each two dimensional vector *(x,y)* the point *(x,y)* in $R^2$, we may say that vectors $(x_1, y_1)$ and $(x_2, y_2)$ are linearly independent if and only if the line passing through the points $(x_1, y_1)$ and $(x_2, y_2)$ does not intersect the origin. In consequence, function *f* is locally injective on *T* if and only if no line simultaneously passes through the origin, $r_1$ and $r_2$ for any $(u,v) \in \tilde{T}$.

Following [Choi-Lee, 1999] we denote by $S_2(c,\delta)$ the region in $R^2$ defined by,

$$
\begin{aligned}
S_2(c,\delta) = \{(x + c_x, y + c_y) / \\
& x = \sum_{i+j+k=3} \delta_{ijk} \frac{\partial}{\partial u} B^3_{ijk}(u,v) \\
& y = \sum_{i+j+k=3} \delta_{ijk} \frac{\partial}{\partial v} B^3_{ijk}(u,v) \\
& c = (c_x, c_y), \quad |\delta_{ijk}| < \delta, \quad (u,v) \in T \ \}
\end{aligned}
$$

Let $e_1 = (1,0)$ and $e_2 = (0,1)$. Then $S_2(e_1, \delta_x)$ is the set of vectors $r_1$ for all 2D triangular Bezier functions with control polygon satisfying $|\Delta x_{ijk}| \le \delta_x$. Analogously, $S_2(e_2, \delta_y)$ is the set of vectors $r_2$ for all 2D triangular Bezier functions with control polygon satisfying $|\Delta y_{ijk}| \le \delta_y$.

Let $\delta_x = \max\left\{ |\Delta x_{ijk}| \right\}$ and $\delta_y = \max\left\{ |\Delta y_{ijk}| \right\}$. Then, for all $(u,v) \in \tilde{T}$ the corresponding vectors $r_1$ and $r_2$ are contained in $S_2(e_1, \delta_x)$ and $S_2(e_2, \delta_y)$ respectively.

Summarizing (as Lemma 1 in [Choi-Lee, 1999]),

**Lemma 1** A 2D triangular Bezier function *f* is locally injective in *T* if no line passes simultaneously through the origin, $S_2(e_1, \delta_x)$ and $S_2(e_2, \delta_y)$.

In the rest of this section we obtain an upper bound for the values of $\delta_x$ and $\delta_y$ such that no line passing through the origin also intersects simultaneously $S_2(e_1, \delta_x)$ and $S_2(e_2, \delta_y)$. Since it is not easy to describe the exact shape of a set $S_2(c,\delta)$, we construct a simple circular region $C_r(c,\delta)$ on the plane containing $S_2(c,\delta)$ and state the upper bounds for $\delta_x$ and $\delta_y$ using circular regions.

Let $C_r(c,\delta)$ be the region in $R^2$ defined by,

$$
C_r(c,\delta) = \{(x + c_x, y + c_y) / x^2 + y^2 \le 72\delta^2\}
$$

where $c=(c_x, c_y)$ .

**Lemma 2** $S_2(c,\delta)\subset C_r(c,\delta)$

**Proof** It is sufficient to show that $S_2(0,\delta)\subset C_r(0,\delta)$ . Let $(x,y)\in S_2(0,\delta)$ , then after the definition of $S_2(0,\delta)$ ,

$$\|(x,y)\|_2^2=\left(\sum_{i+j+k=3}\delta_{ijk}\frac{\partial}{\partial u}B^3_{ijk}(u,v)\right)^2+\left(\sum_{i+j+k=3}\delta_{ijk}\frac{\partial}{\partial v}B^3_{ijk}(u,v)\right)^2$$
$$\leq \delta^2\left(\left(\sum_{i+j+k=3}\left|\frac{\partial}{\partial u}B^3_{ijk}(u,v)\right|\right)^2+\left(\sum_{i+j+k=3}\left|\frac{\partial}{\partial v}B^3_{ijk}(u,v)\right|\right)^2\right)$$

Computing partial derivatives of Bernstein polynomials of degree 3 given by (3) it is straightforward but cumbersome to show that, for $(u,v)\in\tilde{T}$ holds,

$$\sum_{i+j+k=3}\left|\frac{\partial}{\partial u}B^3_{ijk}(u,v)\right|\leq 6$$
$$\sum_{i+j+k=3}\left|\frac{\partial}{\partial v}B^3_{ijk}(u,v)\right|\leq 6$$

In consequence,

$$|(x,y)|_2^2\leq 72\delta^2 \qquad \square$$

**Lemma 3** If a line passes simultaneously through $C_r(e_1,\delta_x)$ , $C_r(e_2,\delta_y)$ and the origin, then $\delta_x^2+\delta_y^2\geq\frac{1}{72}$ .

**Proof** Let $ax+by=0$ be the implicit equation of a line *l* passing simultaneously through $C_r(e_1,\delta_x)$ , $C_r(e_2,\delta_y)$ and (0,0). Denote by $d_1$ and $d_2$ the Euclidean distance from $e_1$ and $e_2$ to the line *l* respectively (see Figure 1 taken from [Choi-Lee, 1999]). Then,

$$d_1=\frac{|a|}{\sqrt{a^2+b^2}},\quad d_2=\frac{|b|}{\sqrt{a^2+b^2}}$$

Therefore, $d_1^2+d_2^2=1$. Since *l* passes through $C_r(e_1,\delta_x)$ , $d_1$ is not greater that the radius of this circle, i.e. $d_1\leq\sqrt{72}\delta_x$ . Similarly, $d_2\leq\sqrt{72}\delta_y$ . Thus,

$$1 = d_1^2 + d_2^2 \le 72(\delta_x^2 + \delta_y^2)$$

Hence,

$$\delta_x^2 + \delta_y^2 \ge \frac{1}{72}$$ □

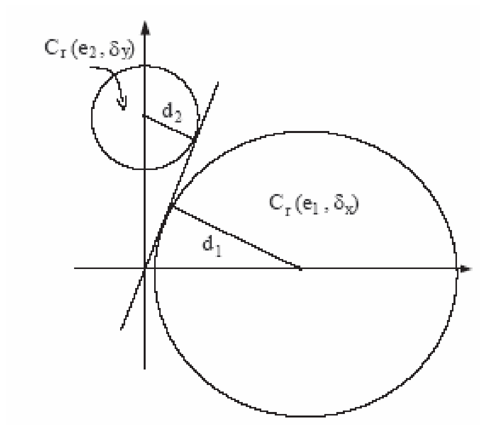



Figure 1: Configuration of $C_r(e_1, \delta_x)$ and $C_r(e_2, \delta_y)$

**Theorem1**: Let $f(u,v) = (x(u,v), y(u,v)) : R^2 \to R^2$ be a 2D triangular cubic Bezier function with *x*(*u*,*v*) and *y*(*u*,*v*) given by (1) and (2). Let $\delta_x = max\{|\Delta x_{ijk}|\}$ and $\delta_y = max\{|\Delta y_{ijk}|\}$ according with (4). If $\delta_x^2 + \delta_y^2 < \frac{1}{72}$ then *f* is locally injective in *T*.

**Proof** After Lemma 3 there is not a line passing simultaneously through $C_r(e_1, \delta_x)$, $C_r(e_2, \delta_y)$ and the origin. Since a $S_2(e_1, \delta_x) \subset C_r(e_1, \delta_x)$ (Lemma 2) and $S_2(e_2, \delta_y) \subset C_r(e_2, \delta_y)$, using Lemma 1 we may conclude that *f* is locally injective in *T*.

□

At first glance, it may seem that the sufficient condition given above is very restrictive concerning the displacement of control points. Nevertheless, due to the affine invariance property of bivariate Bernstein polynomials [Farin, 1993], Theorem 1 could be extended to greater control point displacements. In fact, let $M_2$ be the 2D affine transformation mapping the given control points $b_{ijk}$ as close as possible to the canonical positions $b_{ijk}^0$. Then, instead of applying Theorem 1 to the original control points $b_{ijk}$ we may apply it on the control points $M_2(b_{ijk})$.

### 3.2 Necessary and Sufficient condition

We would like to present another approach, which focuses directly on a necessary and sufficient condition for the local injectivity. As we mentioned above, a function $f(u,v)=(x(u,v),y(u,v))$ is locally injective if and only if the corresponding Jacobian matrix does not vanish on its domain.

Let $C$ be the planar curve,

$$C:\{(u,v)/c(u,v):=x_u y_v - x_v y_u = 0\} \tag{5}$$

Since $f(u,v)$ is a 2D triangular cubic Bezier function, the zero contour $C$ is an implicitly defined plane algebraic curve of degree 4 and can be written using the Bernstein polynomials of the same degree as,

$$C: c(u,v)=\sum_{i+j+k=4} C_{ijk} B^4_{ijk}(u,v)=0 \tag{6}$$

Therefore, $f$ is locally injective if and only if that algebraic curve does not cut the canonical triangle $\tilde{T}$ .

The coefficients $C_{ijk}$ of $C$ in (6) may be computed in terms of the control points $b_{ijk}$ of function $f$ in a very straightforward way as follows. First, we use the expressions for $x_u, x_v, y_u, y_v$ in terms of the Bernstein polynomials and the control points $b_{ijk}$ [Farin, 1993],

$$\begin{aligned}
x_u &= 3\sum_{i+j+k=2} (b^1_{i+1,j,k} - b^1_{i,j,k+1}) B^2_{ijk}(u,v) \\
x_v &= 3\sum_{i+j+k=2} (b^1_{i,j+1,k} - b^1_{i,j,k+1}) B^2_{ijk}(u,v) \\
y_u &= 3\sum_{i+j+k=2} (b^2_{i+1,j,k} - b^2_{i,j,k+1}) B^2_{ijk}(u,v) \\
y_v &= 3\sum_{i+j+k=2} (b^2_{i,j+1,k} - b^2_{i,j,k+1}) B^2_{ijk}(u,v)
\end{aligned} \tag{7}$$

Let $xu_{ijk}, xv_{ijk}, yu_{ijk}$ and $yv_{ijk}$ the coefficients in the Bernstein basis of $x_u, x_v, y_u, y_v$ respectively, which are computed from the previous expressions. For instance, $xu_{020} = 3(b^1_{120} - b^1_{021})$ , and $xv_{020} = 3(b^1_{030} - b^1_{021})$ . Using formula (5) we may compute the coefficients $C_{ijk}$ of $C$ included in the append.

Once we have obtained the implicit form of the curve $C$, we could test if $C$ does not pass through the canonical triangle $\tilde{T}$ , using a convenient range analysis criterion. Usually, range analysis methods (see [Martin et. al., 2002], [Estrada et. al., 2004]) are stated for rectangular regions. Since $C$ is a curve written in barycentric coordinates with respect to the vertices of a triangle, we are forced to develop an adaptation of range analysis for triangular regions.

More precisely, let be $C$ an implicit algebraic curve of degree $n$ written in Bernstein-Bezier form using barycentric coordinates with respect to the vertices of a triangle $T$,

$$C : c(u,v) = \sum_{i+j+k=n} C_{ijk} B_{ijk}^n(u,v) = 0 \tag{8}$$

From the convex hull property of Bernstein polynomials [Farin, 1993], we obtain the following bounds for the values of the function $c(u,v)$ on the triangle $\tilde{T}$ ,

$$min_{ijk}\{C_{ijk}\} \leq \; c(u,v) \leq \; max_{ijk}\{C_{ijk}\}$$

From this fact, we derive the following test:

---

**Test CutTriangle** $(C_{ijk}, T)$

**Compute:** $k_{\min} := min_{ijk}\{C_{ijk}\}$ **,** $k_{\max} := \max_{ijk}\{C_{ijk}\}$

**if** $0 \notin [k_{\min}, k_{\max}]$ **then**

    **return false** (C does not pass through the triangle T)

**else**

    **return true** (C might pass trough the triangle T)

---

In order to render the algebraic curve *C* we propose the algorithm **GetTriangles** based on subdivision and range analysis for a triangular region. The algorithm may be described as follows. First we check if *C* does not pass through *T* using **Test CutTriangle**. If **CutTriangle** answer is false we finish. Otherwise, subdivide $\tilde{T}$ in 4 triangular regions $T_l, \quad l = 1,...,4$. For each index *l* we compute the coefficients of *C* restricted to the triangle $T_l$ and apply recursively the above procedure, until the length of the equal size sides of $T_l$ is smaller than some prescribed tolerance $\varepsilon$. As a result of this algorithm we obtain a set of triangles where the curve might intersect the triangle *T*. There is a bijective correspondence among this set of triangles in *T* and a set of triangles in screen coordinates. Choosing $\varepsilon$ small enough we can get a set of triangles in screen coordinates, such that the largest side length is of order of the length of a pixel side. Drawing the pixel containing the barycenter of each triangle of this last set, we may render the graph of *C* on *T*.

---

**Algorithm 3.1** Generating triangles where a curve *C* in Bernstein form intersects a triangle *T*.

---

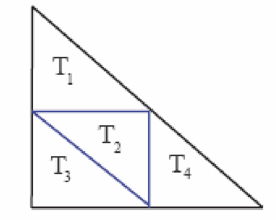
**Test GetTriangles** $(C_{ijk}, T, \varepsilon)$

**if CutTriangle** $(C_{ijk}, T)$ **= true then**

    **if** $size(T) \leq \varepsilon$ **then**

        **output T**

    **else**




**for** $l = 1,...,4$

**Compute** $C^l_{ijk}$ **=** Coefficients of C restricted to $T_l$

**GetTriangles** $(C_{ijk}, T_l, \varepsilon)$

**else**

**return**

To compute the coefficients $C^{\,l}_{ijk}$ of *C* restricted to $T_l$ we use the blossom principle for the 2D triangular Bezier function *c*(*u*,*v*) given in (8). Let $\boldsymbol{f}_{1l}, \boldsymbol{f}_{2l}, \boldsymbol{f}_{3l}$ be the barycentric coordinates of the vertices of the triangle $T_l$ contained in $\tilde{T}$ . Then [Farin, 1993],

$$C^l_{ijk} = \boldsymbol{b}\,[\,\boldsymbol{f}_{1l}^{<i>}, \boldsymbol{f}_{2l}^{<j>}, \boldsymbol{f}_{3l}^{<k>}\,]$$

where ***b*** represents the blossom, $\boldsymbol{f}_{1l}^{<i>}$ is short for *i*-fold repetition of $\boldsymbol{f}_{1l}$ and similar notation have been used for *j*-fold repetition of $\boldsymbol{f}_{2l}$ and for *k*-fold repetition of $\boldsymbol{f}_{3l}$ (see [Farin, 1993]).

To finish this section, we describe the algorithm **InjecTest** to determine if the 2D triangular cubic Bezier function *f* given by (1) and (2) is locally injective in a triangle *T*. First, the coefficients of the partial derivatives are computed using (7). Then the coefficients of the curve *c*(*u*,*v*)=0 are obtained from the formulas in the Append. Finally, the algorithm **GetTriangle** is used to check if the plane algebraic curve (6) cuts the triangle *T*.

---

**Algorithm 3.2** Local injectivity test for the 2D triangular cubic Bezier function *f* given by (1) and (2) in a triangle *T*.

---

**InjecTest** $(f, T, \varepsilon)$

- Compute the coefficients of the partial derivatives of *f* using (7).
- Compute the coefficients of the curve *c(u,v)=0* from the formulas in the Append
- Get the list *L* of triangles where the curve *c(u,v)=0* intersects the triangle *T* using algorithm **GetTriangles** $(C_{ijk}, T, \varepsilon)$
- **if** IsEmpty (L) = **true then**

  **Output:** *f* is locally injective on *T*

  **else**

  **Output:** *f* is not locally injective in a neighborhood of the triangles in *L*

As result of this algorithm, one obtains either a certificate for the local injectivity of *f* on the interior of *T*, or a set of triangles where the curve *C* might cut *T*.

## 4 EXAMPLES

### Example 1

Suppose that *f*(*u*,*v*)=(*x*(*u*,*v*),*y*(*u*,*v*)) is the linear function,

$$x(u,v) = u + v$$
$$y(u,v) = u - v$$

Then, the control net of function *f*(*u*,*v*) has coordinates $b_{ijk} = ((i+j)/3, (i-j)/3)$ with $i + j + k = 3$. Due to linearity of *f*, functions $x_u, x_v, y_u$ and $y_v$ are constants. Even more,

$$c(u,v) = -2$$

In consequence *C* does not cut the canonical triangle $\tilde{T}$ and we conclude that *f* is locally injective in $\tilde{T}$ .

### Example 2

Let *f*(*u*,*v*)=(*x*(*u*,*v*),*y*(*u*,*v*)) the 2D triangular cubic function given by,

$$x(u,v) = 2(1-u-v)^3 + 6uv(1-u-v)$$
$$y(u,v) = (1-u-v)^3 + 6uv(1-u-v) + u^3$$

We construct a regular grid of points $(u_i, v_i)$ on the canonical triangle and compute their image $f(u_i, v_i)$. The results are shown in Figure 2. As we observe, there are some regions of $f(\tilde{T})$ where points $f(u_i, v_i)$ are very concentrated. In consequence, we suspect that *f* is not injective in $\tilde{T}$ .

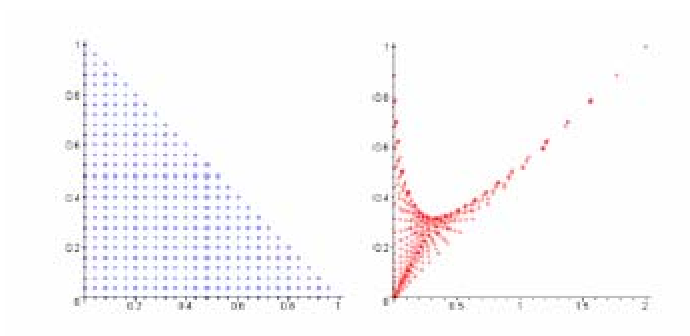


Figure 2: A regular grid on $\tilde{T}$ and its image by *f*(*u*,*v*)

For instance, in a neighborhood of the point $(0.28, 0.32) \in f(\tilde{T})$, the grid of points is very dense. Let´s consider the points $(u,v) \in \tilde{T}$, such that $x(u,v) = 0.28, y(u,v) = 0.32$. As shown in Figure 3, points $p_0 = (u_0, v_0) = (0.42, 0.24)$ and $p_1 = (u_1, v_1) = (0.46, 0.16)$ are on the intersection of the isolines $x(u,v) - 0.28 = 0, \quad y(u,v) - 0.32 = 0$, hence *f* is not injective.

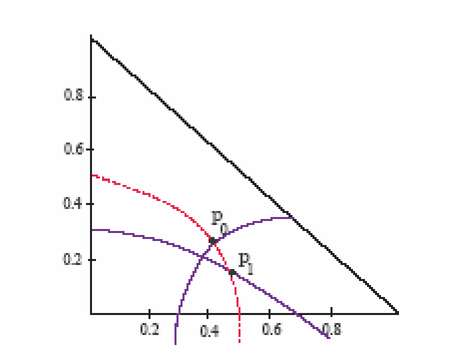



Figure 3: The isolines $x(u,v) - 0.28 = 0, \quad y(u,v) - 0.32 = 0$

From the expressions for *x*(*u*,*v*) and *y*(*u*,*v*) we may compute directly the coefficients $C_{ijk}$ of the curve *c*(*u*,*v*). Of course, the same result is obtained if we use formulas for $C_{ijk}$ contained in the append,

$$c(u,v) = -\frac{9}{2} B^4_{103}(u,v) + 3B^4_{202}(u,v) - \frac{9}{2} B^4_{301}(u,v) + \frac{9}{2} B^4_{310}(u,v) + \frac{9}{2} B^4_{013}(u,v) = 0 \qquad (9)$$

Figure 4 shows the graph of the implicit curve (9) in the canonical triangle. Obviously, *f* is not locally injective for any point on this curve. In particular, observe that $c(p_0) = 0.11$ and $c(p_1) = -0.13$, hence both points are very close to a point on the curve *C* in the interior of $\tilde{T}$.

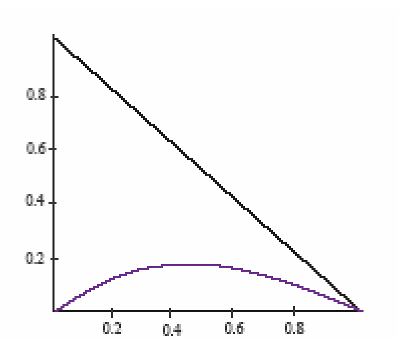



Figure 4: Graph of the implicit algebraic curve (8) in the canonical triangle.

## 5 NUMERICAL EXPERIMENTS

The results included in this section were obtained implementing our algorithm **InjecTest** in the C++ language and using the graphical library OpenGL. The coefficients of the curve *C* were computed following the formulas for $C_{ijk}$ presented in the append. Additionally, the program includes an implementation of the algorithm **GetTriangles** to get a sample of points on the curve *C* contained in the canonical triangle $\tilde{T}$ .

**Example 1**

Let $f(u,v)=(x(u,v),y(u,v))$ the 2D triangular cubic Bezier function,

$$x(u,v)=B_{003}^{3}(u,v)=(1-u-v)^{3}$$
$$y(u,v)=B_{021}^{3}(u,v)=3v^{2}(1-u-v)$$

Computing the partial derivatives $x_u, x_v, y_u$ and $y_v$ of $x(u,v)$ and $y(u,v)$ respectively we obtain the equation of the curve *C*,

$$c(u,v)=-\frac{9}{2}B_{013}^{4}(u,v)=0$$

Its graph is shown in Figure 5. From the expression of the curve *c*(*u*,*v*)=0 we may conclude that the function *f*(*u*,*v*) is injective in all points (*u*,*v*) inside the canonical triangle, except perhaps in a neighborhood of the points $(u,0)$ and $(u,1-u)$ having $u \in [0,1]$.

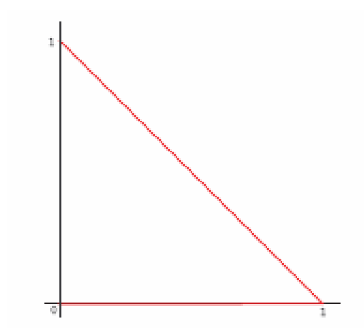



Figure 5: $c(u,v)=-18v(1-u-v)^{3}=0$

In Figure 6 the image $T^{'}$ of a triangulation *T* of the canonical triangle is computed using the application *f*. Since *f* has the value (0,0) for any point (*u*,1-*u*), the image triangles $T_5^{'}, T_8^{'}$ and $T_9^{'}$ are degenerated.

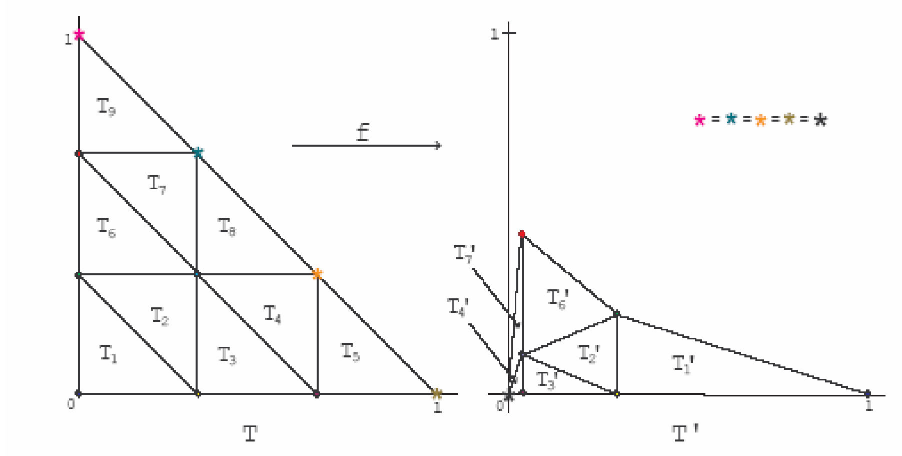



Figure 6: The image $T^{'}$ of the triangulation *T* by the application *f*. The triangles $T_5^{'}, T_8^{'}$ and $T_9^{'}$ are degenerated.

**Example 2**

Let $f(u,v) = (x(u,v), y(u,v))$ the 2D triangular cubic function given by,

$$x(u,v) = \frac{1}{8}B_{102}^{3}(u,v) + \frac{3}{8}B_{201}^{3}(u,v) + \frac{1}{8}B_{111}^{3}(u,v) + B_{300}^{4}(u,v) + \frac{3}{8}B_{210}^{3}(u,v) + \frac{1}{8}B_{120}^{3}(u,v)$$
$$= \frac{1}{8}u(3 + 3u + 2u^{2})$$

$$y(u,v) = \frac{1}{8}B_{012}^{3}(u,v) + \frac{1}{8}B_{111}^{3}(u,v) + \frac{3}{8}B_{021}^{3}(u,v) + \frac{1}{8}B_{210}^{3}(u,v) + \frac{3}{8}B_{120}^{3}(u,v) + B_{030}^{3}(u,v)$$
$$= \frac{1}{8}v(3 + 3v + 2v^{2})$$

The equation of the curve *c*(*u*,*v*)=0 is given in (10). Since all the Bernstein coefficients of *C* are positive,

$$\begin{aligned} c(u,v) = {} & \frac{9}{64}B_{004}^{4}(u,v) + \frac{27}{128}B_{103}^{4}(u,v) + \frac{27}{128}B_{013}^{4}(u,v) + \frac{21}{64}B_{202}^{4}(u,v) + \frac{21}{64}B_{112}^{4}(u,v) + \\ & + \frac{21}{64}B_{022}^{4}(u,v) + \frac{63}{128}B_{301}^{4}(u,v) + \frac{69}{128}B_{211}^{4}(u,v) + \frac{69}{128}B_{121}^{4}(u,v) + \frac{63}{128}B_{031}^{4}(u,v) + \\ & + \frac{45}{64}B_{400}^{4}(u,v) + \frac{27}{32}B_{310}^{4}(u,v) + \frac{63}{64}B_{220}^{4}(u,v) + \frac{27}{32}B_{130}^{4}(u,v) + \frac{45}{64}B_{040}^{4}(u,v) \end{aligned} \tag{10}$$

the curve does not cut the canonical triangle $\tilde{T}$. In consequence, the function *f*(*u*,*v*) is injective inside of $\tilde{T}$.

**Example 3**

Let $f(u,v) = (x(u,v), y(u,v))$ the 2D triangular cubic function given by,

$$x(u,v) = B^3_{003}(u,v) - B^3_{021}(u,v) = (1-u-v)^3 - 3v^2(1-u-v)$$
$$y(u,v) = B^3_{003}(u,v) + B^3_{021}(u,v) - B^3_{300}(u,v) = (1-u-v)^3 + 3v^2(1-u-v) - u^3$$

Computing the partial derivatives $x_u, x_v, y_u$ and $y_v$ of *x* and *y* it is easy to check that the curve *c*(*u*,*v*)=0 is given by,

$$c(u,v) = -9B^4_{013}(u,v) - \frac{3}{2}B^4_{202}(u,v) - \frac{3}{2}B^4_{211}(u,v) + \frac{3}{2}B^4_{220}(u,v) \tag{11}$$

The graph of the curve (11) obtained using the algorithm **GetPoints** is shown in the last picture of Figure 7. Also some intermediate steps of this algorithm are shown in Figure 7 in order to illustrate the performance of the algorithm. The conclusion is that since the curve *C* intersects the canonical triangle the function *f*(*u*,*v*) is not injective inside it.

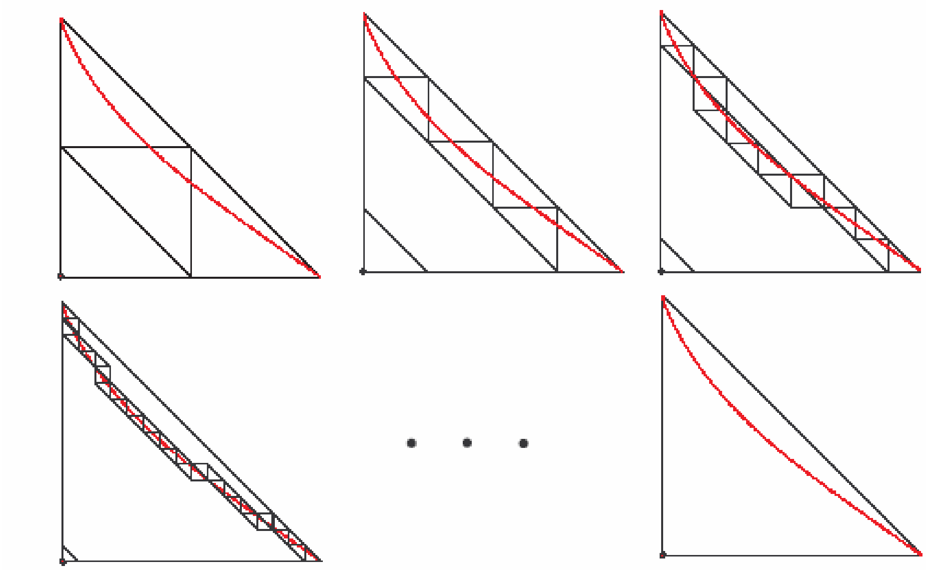


Figure 7: Some intermediate subdivision steps in algorithm **GetTriangles** for the algebraic curve given by (10).

## 6 APPEND

In order to obtain the coefficients $C_{ijk}, i+j+k=4$ of the curve *C* given by (6) we take into account that if $B^2_{r_1,r_2,r_3}, B^2_{s_1,s_2,s_3}$ are Bernstein polynomials of degree 2 with $r_1+r_2+r_3=2,\quad s_1+s_2+s_3=2$ and $r_1+s_1=i,\quad r_2+s_2=j$ and $r_3+s_3=k$ then,

$$B^2_{r_1,r_2,r_3}B^2_{s_1,s_2,s_3} = \frac{1}{6}\frac{i!\,j!\,k!}{r_1!\,r_2!\,r_3!\,s_1!\,s_2!\,s_3!}B^4_{ijk}(u,v)$$

where $B^4_{ijk}(u,v)$ are Bernstein polynomials of degree 4.

In consequence it is straightforward to check that,

$$C_{ijk} = \sum_{|r|=2} \sum_{|s|=2} \sum_{|r+s|=(ijk)} l_{ijk} (xu_{r1,r2,r3} yv_{s1,s2,s3} - xv_{r1,r2,r3} yu_{s1,s2,s3})$$

where $l_{ijk} = \frac{1}{6} \frac{i!\,j!\,k!}{r_1!\,r_2!\,r_3!\,s_1!\,s_2!\,s_3!}$ and $r = (r_1, r_2, r_3)$, $|r| = r_1 + r_2 + r_3$ and similarly for *s* and |*s*|.

## REFERENCES


BUCK, R. C. [1978]: **Advanced Calculus**, McGraw Hill, N. York. 3rd Edition.

CHOI, Y. and LEE, S. [1999]: **Local injectivity conditions of 2D and 3D uniform cubic B-spline functions**, Proceedings of Seventh Pacific Conference on Computer Graphics and Applications, 302–312.

ESTRADA J., MARTÍNEZ D., LEÓN D. THEISEL H. [2004]: **Solving Geometric Problems using Subdivision Methods and Range Analysis**, Mathematical Methods for Curves and Surfaces, 101–114, Nashboro Press, Brentwood, TN.

FARIN, G. [1993]: **Curves and Surfaces for CAGD: A practical Guide**, Academic Press, N. York. 3rd Edition.

FLOATER, M. [1997]:Parameterization and smooth approximation of surface triangulation, **Computer Aided Geometric Design** 14, 231–250.

FLOATER, M. AND C. GOSTMAN [1999]: How to morph tilings injectively, **J. of Computational and Applied Mathematics** 101, 117–129.

FLOATER, M. [2003]: One-to-one piecewise linear mappings over triangulations, **Math. Comp.** 72, 685–696.

FLOATER, M. and K. HORMAN [2004]: **Surface Parameterization: a Tutorial and Survey**, Advances on Multiresolution *in Geom. Modelling, N. Dodgson, M. Floater, M. Sabin (eds.)*, Springer-Verlag, Berlin157–186.

FUJIMURA. K. and M. MAKAROV [1998]: Foldover-free image warping, **Graphical Models and Image Proccesing** 2, 100–111.

GOODMAN, T. and UNSWORTH, K. [1996], Injective bivariate maps, **Ann. Numer. Math**. 3, 91–104.

LEE, S., CHWA, K. Y., SHIN, S. Y. AND G. WOLBERG [1995]: Image metamorphosis using snakes and free-form deformations, **Proceedings of SIGGRAPH** 95, 439–448.

LEE, S., CHWA, K. Y., HAHN, J. and S. Y. SHIN [1996]: Image morphing using deformation techniques, **J. Visualization and Computer Animation,** 4, 3–23.

LEE, S., WOLBERG, G., CHWA, K. Y. and S. Y. SHIN [1996]: Image metamorphosis with scattered feature constrains**, IEEE Trans. Visualization and Computer Graphics**, 4, 337–354.

MARTIN, R., SHOU, H., VOICULESCU,I., BOWYER, A. and WANG, G. [2002]: Comparison of interval methods for plotting algebraic curves, **CAGD** 19, 553–581.

SEDENBERG, T. W. and S. R. Parry [1986]: **Free-form deformation of solid geometric models**, Proceedings of SIGGRAPH 86, 151–160.

WOLBERG, G.[1990]: **Digital Image Warping**, IEEE Computer Society Press, Los Alamitos, CA.