A Strategy for the Interpolation of Surfaces Through the Use of Basis Functions

EDSON RICARDO DE ANDRADE SILVA[1], CRETO AUGUSTO VIDAL[1], JOAQUIM BENTO CAVALCANTE NETO[1]

[1]UFC–Universidade Federal do Ceará - Bloco 910, Campus do Pici, Fortaleza, CE, Brasil
{latino, cvidal, joaquimb}@lia.ufc.br

Abstract. For the construction of digital terrain models based on surface interpolation, it is defined a bivariate function $F(x, y)$ that interpolates a finite set of N sample points, $P_i = (x_i, y_i, z_i)$, such that, $F(x_i, y_i) = z_i$. In this work, it is presented a strategy for the generation of interpolation surfaces through the use of basis functions. This methodology is based on a work by Chaturvedi and Piegl, where improvements related to the construction of the basis functions were made. The proposed strategy allows a larger expansion of the basis function's support region, represented by the interior of a trajectory curve, composed of quadratic rational Bézier segments and reduces the approximation error between the reference surface and the interpolation surface.

1 Introduction

In Digital Terrain Modeling, the concept of height fields is mathematically represented by an elevation function $H : D \subset R^2 \rightarrow R$, defined on the domain D of a reference plane. Usually, the elevation values are known only at a finite set of sample points.

The use of irregularly distributed sample points avoids the over sampling problem that is common to Data Elevation Models [1], where the sample points are distributed over a regular grid.

Regardless of the type of sample point distribution used, a way of computing the elevation at any point $\mathbf{X}$ of the domain is needed. For that purpose, several surface interpolation strategies can be found in the literature, which guarantee that the Digital Terrain Model has: Shape Fidelity, Domain Independence, Locality and Editability [1]-[7].

In this work, it is presented a strategy for the generation of interpolation surfaces through the use of basis functions. This methodology is based on a work of Chaturvedi and Piegl [5], where improvements related to the construction of the basis functions were made. This methodology allows a larger expansion of the basis function's support region, represented by the interior of a trajectory curve, composed of quadratic rational Bézier segments. Moreover, it promotes a better agreement between the interpolating surface and a reference terrain.

The surface is defined procedurally, based on a triangulation of the projections, $\mathbf{P}'_k = (x_k, y_k)$, of the sample points, $\mathbf{P}_k = (x_k, y_k, z_k)$, $1 \leq k \leq N$, on a reference plane xy. The interpolation surface is constructed by a suitable combination of N specially defined basis functions, corresponding to the sample points.

Each basis function is defined on a local domain in the neighborhood of the corresponding projection $\mathbf{P}'_k$. This domain includes only $\mathbf{P}'_k$ and its adjacent points in the triangulation. The illustration in Figure 1 shows two distinct basis functions associated with the sample point $\mathbf{P}'_k$: one, corresponding to a piecewise linear interpolation; and the other, constructed over an expanded domain and utilizing a smooth profile curve.

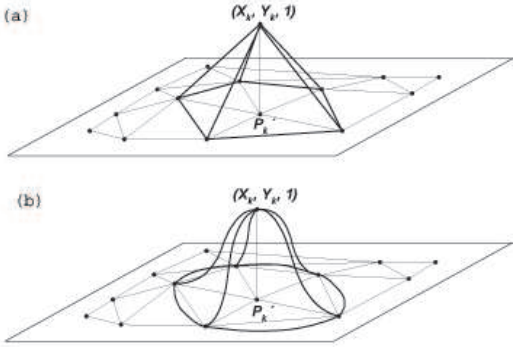

Figure 1: Two possible basis functions.

The smoothness of the profile curve and the extension of the domain are controlled by local parameters β and $\alpha \in [0, 1]$, respectively, which influences the overall shape of the basis function.

The remainder of the paper is organized as follows. In Section 2, it is presented an overview of the surface reconstruction problem, using basis functions. In Section 3, it is discussed the aspects of construction and control of basis functions. In Section 4, the proposed method is contrasted against other methods. Finally, some conclusions are drawn in Section 5.

2 Interpolation Surface

The surface reconstruction problem of a terrain, for which a set of sample points $\mathbf{P}_k = (x_k, y_k, z_k) \in R^3$, $k =$

$1, 2, ..., N$ is known, consists in determining a bivariate function $S(x, y)$, such that $S(x_k, y_k) = z_k$ for each point $\mathbf{P}'_k \in D \subset R^2$. The points $\mathbf{P}'_k$ are the orthogonal projections of points $\mathbf{P}_k$ on the reference plane xy.

Equation (1) determines the interpolation surface $S(x, y)$ by a weighted sum of N basis functions $W_k(x, y)$. Thus,

$$S(x, y) = \sum_{k=1}^{N} z_k W_k(x, y). \quad (1)$$

In digital terrain modeling:

1. z_k are the weights;
2. $W_k(x, y) \geq 0, \forall (x, y) \in D$ (non-negativity);
3. $\sum_{k=1}^{N} W_k(x, y) = 1, \forall (x, y) \in D$ (partition of unity);
4. $W_k(x_j, y_j) = \delta_{kj}$, where $\delta_{kj} = 0$ if $k \neq j$ and $\delta_{kj} = 1$ otherwise;
5. $W_k(x, y) = 0, \forall (x, y) \notin D_k$, where D_k is the support domain of W_k.

As pointed out in [5], requirements 2-5 are easily satisfied if rational basis functions of the form

$$W_k(x, y) = (w_k(x, y))/(\sum_{j=1}^{N} w_j(x, y)) \quad (2)$$

are used. It is also pointed out, in the same work, that these basis functions are analogous to the tensor product B-spline basis functions in that: 1) they can be defined over arbitrary domains (concave or convex); 2) they are identically zero outside their support domains; 3) they are positive within their support domains and attain a single maximum; and 4) they do not possess local maxima or minima.

The methods for constructing the interpolation surface $S(x, y)$ based on Equation (1) are distinguishable, both, by the way the basis functions $W_k(x, y)$ are defined, and by the way the basis functions' support domains are delimited. For example, the basis functions of the polyhedral interpolation are swung surfaces defined by rotating a linear profile curve around a local z-axis and scaling the profile by the boundary polygon of the support region. This region is identified by the triangulation of the points $\mathbf{P}'_k$ (Figure 1(a)).

In summary, given an arbitrary point $(\hat{x}, \hat{y})$, to obtain $\hat{z} = S(\hat{x}, \hat{y})$ it is necessary: 1) to find all the basis functions that do not vanish at $(\hat{x}, \hat{y})$; 2) to evaluate each non-zero contribution of these basis functions by, first, computing $w_k(\hat{x}, \hat{y})$ and, then, substituting into Equation (2); and 3) to blend the basis functions values with the corresponding sample points, using Equation (1).

3 Basis Functions

In this section, it is discussed a two-step process for the construction of basis functions. First, a trajectory curve that delimits the support domain of the basis function is determined. Then, a profile curve is defined to generate a swung surface as it is scaled by the trajectory curve during its revolution around the corresponding sample point projector. The main improvements over the process proposed in [5] are made in the construction of the trajectory curves.

3.1 Trajectory Curve

In order to determine the trajectory curve that delimits the support domain of the basis function associated with a sample point $\mathbf{P}_k$, one starts by obtaining a triangulation of the projections, $\mathbf{P}'_j = (x_j, y_j)$, of the N sample points on the reference plane. Next, using all the triangles that share the vertex $\mathbf{P}'_k$, a polygon is defined by connecting the sides opposite to that vertex. Finally, it is constructed a sequence of NURBS (*Non-Uniform Rational B-Splines*)[9] relative to each side of the polygon just defined such that they maintain C^1 continuity at the connecting points.

The resulting trajectory curve can be either open, whenever $\mathbf{P}'_k$ is at the border of the terrain's domain D; or closed, when it is an interior point of D (See Figure 2). In synthesis, the trajectory curve must: 1) interpolate all the vertices of the polygon used to define the support region; 2) possess, at least, C^1 continuity ; 3) define a region in which $\mathbf{P}'_k$ is the only sample point projection strictly inside it; and, 4) be intersected at most once by any straight-line segment with origin at point $\mathbf{P}'_k$.

Conditions 1) and 2) are trivially satisfied, if NURBS segments are chosen to connect each two consecutive vertices of the polygon. Condition 3) implies that points $\mathbf{P}'_j$ which are not part of the boundary polygon impose limits on the expansion of the trajectory curve, since none of them can lie inside the support region. Condition 4) imposes a special strategy for the construction of the trajectory curve, as discussed next.

The NURBS segments chosen are quadratic rational Bézier segments defined by

$$\mathbf{C}(v) = \frac{w_0 \mathbf{b}_0 B_0^2(v) + w_1 \mathbf{b}_1 B_1^2(v) + w_2 \mathbf{b}_2 B_2^2(v)}{w_0 B_0^2(v) + w_1 B_1^2(v) + w_2 B_2^2(v)}, \quad (3)$$

where $\mathbf{b}_0$, $\mathbf{b}_1$, and $\mathbf{b}_2$ are the control points; w_i is the i-th weight and $B_i^2(v)$ is the i-th second degree Bernstein's polynomial.

The curve $\mathbf{C}(v)$ interpolates the points $\mathbf{b}_0$ and $\mathbf{b}_2$, and, at these points, it is tangent to the vectors $\mathbf{b}_1 - \mathbf{b}_0$ and $\mathbf{b}_2 - \mathbf{b}_1$, respectively. For the purposes of implementation, one can set $w_0 = w_2 = 1$ and vary w_1 (denoted by w in the following). As w increases, the curve approaches point $\mathbf{b}_1$.

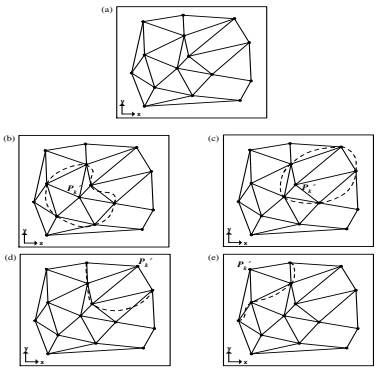

Figure 2: Types of trajectory curves. (a) and (b) closed trajectory curves; (c) and (d) open trajectory curves.

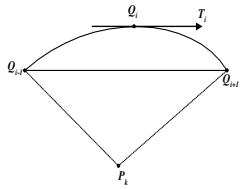

Figure 3: Computation of the tangent $\mathbf{T}_i$ at $\mathbf{Q}_i$.

Suppose that $\{\mathbf{Q}_1, \mathbf{Q}_2, ..., \mathbf{Q}_n\}$ is the set of vertices of the polygon around point $\mathbf{P}'_k$ determined by the triangulation (see Figure 2), where $\mathbf{Q}_n = \mathbf{Q}_1$ when the curve is closed. Between each pair of vertices $(\mathbf{Q}_i, \mathbf{Q}_{i+1})$, one tries to define a connecting quadratic rational Bézier segment. However, that is not always possible, requiring that more than one segment be used.

In order to determine the connecting NURBS segments, it is necessary to find the tangent vectors $\mathbf{T}_i$ to the trajectory curve at the vertices $\mathbf{Q}_i$. A simple way to estimate these tangent vectors is to use the central difference scheme represented by the equation

$$\mathbf{T}_i = (\mathbf{Q}_{i+1} - \mathbf{Q}_{i-1})/(|\mathbf{Q}_{i+1} - \mathbf{Q}_{i-1}|), \qquad (4)$$

where $\mathbf{T}_i$ is an unit vector with the same orientation as vector $\mathbf{Q}_{i+1} - \mathbf{Q}_{i-1}$(Figure 3).

Equation (4) adequately defines the tangent at all the points $\mathbf{Q}_i$ whenever the trajectory curve is closed. However, when that curve is open, it is necessary to define another way of computing the tangent vectors at the extremities $\mathbf{Q}_1$ and $\mathbf{Q}_n$. Hence:

1. If $n = 2$, $\mathbf{T}_1$ and $\mathbf{T}_2$ are defined as the unit vectors that form 45^o angles with the vector $\mathbf{Q}_2 - \mathbf{Q}_1$ (Figure 4(a));

2. If $n \geq 3$, $\mathbf{T}_1$ is defined as the unit vector tangent to the parabola passing through the vertices $\mathbf{Q}_1$, $\mathbf{Q}_2$ and $\mathbf{Q}_3$ at vertex $\mathbf{Q}_1$; and $\mathbf{T}_n$, as the unit vector tangent to the parabola passing through the vertices $\mathbf{Q}_{n-2}$, $\mathbf{Q}_{n-1}$ and $\mathbf{Q}_n$ at vertex $\mathbf{Q}_n$. However, in order to guarantee that all straight-line segments with origin at $\mathbf{P}'_k$ do not intersect the trajectory curve more than once, one computes the angles θ_i between the vectors $\mathbf{R}_i = \frac{\mathbf{Q}_i - \mathbf{P}'_k}{|\mathbf{Q}_i - \mathbf{P}'_k|}$ and $\mathbf{T}_i$ $(i = 1, n)$, measured clockwise (Figure 4(b)) and, in case $180^o < \theta_i < 270^o$, $\mathbf{T}_i = -\mathbf{R}_i$ or, in case $\theta_i > 270^o$, $\mathbf{T}_i = \mathbf{R}_i$.

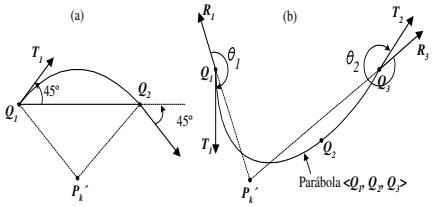

Figure 4: Tangent at the ends of the trajectory curve.

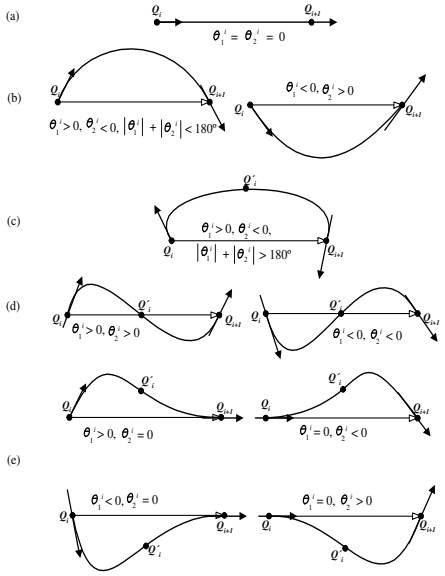

Figure 5: Configurations of the trajectory curves.

Therefore, the problem of computing the trajectory curve passing through the vertices $\mathbf{Q}_1, \mathbf{Q}_2, ..., \mathbf{Q}_n$, is restricted to determining the control points of each Bézier segment connecting $\mathbf{Q}_i$ to $\mathbf{Q}_{i+1}$, since the tangent at these vertices are known.

If the tangents were "well behaved", as shown in Figure 5(b), one would simply choose the intersection point of the parameterized straight-lines $\mathbf{Q}_i + s\mathbf{T}_i$ and $\mathbf{Q}_{i+1} + t\mathbf{T}_{i+1}$ as the middle control point, $\mathbf{b}_1$. However, due to the variety of shapes that the trajectory curve may assume, not always

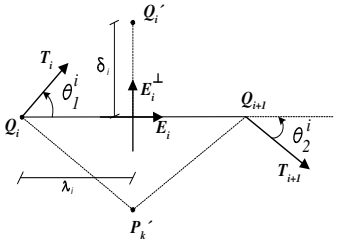

Figure 6: Geometrical interpretation of λ_i and δ_i.

it can be represented by a single quadratic Bézier segment. For example, this is the case of the curve segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$ in Figure 5(d), since a quadratic rational Bézier curve cannot present the inflections shown in that figure.

Now, consider θ_1^i and θ_2^i the angles that the tangent vectors $\mathbf{T}_i$ and $\mathbf{T}_{i+1}$, respectively, form with the vector $\mathbf{Q}_{i+1} - \mathbf{Q}_i$. Also, consider s_1^i and s_2^i the signs of these angles, i.e., $s_1^i = 1$ if $(\mathbf{Q}_{i+1} - \mathbf{Q}_i) \times \mathbf{T}_i > 0$, $s_1^i = -1$ if $(\mathbf{Q}_{i+1} - \mathbf{Q}_i) \times \mathbf{T}_i < 0$, and $s_1^i = 0$ if $(\mathbf{Q}_{i+1} - \mathbf{Q}_i) \times \mathbf{T}_i = 0$. s_2^i is defined, analogously, by examining the sign of the vector product $(\mathbf{Q}_{i+1} - \mathbf{Q}_i) \times \mathbf{T}_{i+1}$. The strategy for determining the control points is defined by analyzing the five cases depicted in Figure 5 and described as follows.

Case 1 ($s_1^i = s_2^i = 0$) (Figure 5 (a)): The Bézier segment consists of a straight-line segment. One chooses $\mathbf{b}_0 = \mathbf{Q}_i$, $\mathbf{b}_1 = (\mathbf{Q}_i + \mathbf{Q}_{i+1})/2$ and $\mathbf{b}_2 = \mathbf{Q}_{i+1}$.

Case 2 ($s_1^i s_2^i = -1$ and $\left|\theta_1^i\right| + \left|\theta_2^i\right| < 180^o$) (Figure 5(b)): One chooses $\mathbf{b}_0 = \mathbf{Q}_i$, $\mathbf{b}_1$ as the intersection of the parameterized straight-lines $\mathbf{Q}_i + s\mathbf{T}_i$ and $\mathbf{Q}_{i+1} + t\mathbf{T}_{i+1}$ and $\mathbf{b}_2 = \mathbf{Q}_{i+1}$.

In the next three cases, two Bézier segments are used to interpolate the curve segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$. Therefore, it is necessary, first, to include a new point $\mathbf{Q}_i'$ and only then to interpolate the curve segments $\mathbf{Q}_i\mathbf{Q}_i'$ e $\mathbf{Q}_i'\mathbf{Q}_{i+1}$. Point $\mathbf{Q}_i'$ is defined by

$$\mathbf{Q}_i' = \mathbf{Q}_i + \lambda_i \mathbf{E}_i + \delta_i \mathbf{E}_i^{\perp}, \tag{5}$$

where $\lambda_i = |\mathbf{Q}_{i+1} - \mathbf{Q}_i|/2$, $\mathbf{E}_i$ is the unit vector in the direction of vector $\mathbf{Q}_{i+1} - \mathbf{Q}_i$, $\mathbf{E}_i^{\perp}$ is the unit vector perpendicular to $\mathbf{E}_i$ and δ_i is the distance from point $\mathbf{Q}_i'$ to the chord $\mathbf{Q}_i\mathbf{Q}_{i+1}$ (Figure 6). Moreover, for each case, different δ_i and $\mathbf{T}_i'$ (tangent vector at $\mathbf{Q}_i'$) are computed.

Case 3 ($s_1^i s_2^i = -1$ and $\left|\theta_1^i\right| + \left|\theta_2^i\right| > 180^o$) (Figure 5(c)): One computes $\delta_i = \min(d_i, d_\alpha)$, where d_i is the distance from the closest non-neighbor point $\mathbf{P}_i'$ to the chord $\mathbf{Q}_i\mathbf{Q}_{i+1}$ and d_α is a chosen fraction of this chord segment, i.e. , $d_\alpha = \alpha\,|\mathbf{Q}_{i+1} - \mathbf{Q}_i|$. The tangent $\mathbf{T}_i'$ at $\mathbf{Q}_i'$ is the vector parallel to the chord $\mathbf{Q}_i\mathbf{Q}_{i+1}$.

Case 4 ($s_1^i s_2^i = 1$) (Figure 5(d)): Taking $\delta_i = 0$, one determines the point $\mathbf{Q}_i'$ on the chord segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$. The tangent at $\mathbf{Q}_i'$ is then defined as

$$\mathbf{T}_i' = cos(\theta_i')\mathbf{E}_i + sin(\theta_i')\mathbf{E}_i^{\perp}, \tag{6}$$

where

$$\theta_i' = -\frac{3}{4} s_1^i \min(\min(\left|\theta_1^i\right|, \pi - \left|\theta_1^i\right|), \min(\left|\theta_2^i\right|, \pi - \left|\theta_2^i\right|)). \tag{7}$$

In this case, one of the Bézier segments relative to the segments $\mathbf{Q}_i\mathbf{Q}_i'$ and $\mathbf{Q}_i'\mathbf{Q}_{i+1}$ is located in the same semi-plane as $\mathbf{P}_k'$ relative to the chord segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$ and, therefore, it is necessary a correction strategy, as shown in Figure 4(b).

Case 5 ($s_1^i s_2^i = 0$ and $\left|s_1^i\right| + \left|s_2^i\right| > 0$) (Figure 5(e)): One computes the scalar δ_i by

$$\delta_i = \frac{1}{2}(s_1^i - s_2^i)\min\left(d_p,\, d_\alpha,\, 2\,|\beta_i - \lambda_i|\, tan(|\theta_i'|)\right), \tag{8}$$

where

$$d_p = \begin{cases} d_k, \text{ if } (s_1^i - s_2^i) < 0 \\ d_i, \text{ if } (s_1^i - s_2^i) > 0 \end{cases}, \tag{9}$$

d_k is the distance from point $\mathbf{P}_k'$ to the chord segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$,

$$\beta_i = \begin{cases} \frac{1}{4}\left|\mathbf{Q}_{i+1} - \mathbf{Q}_i\right|, \text{ if } \left|s_1^i\right| = 0 \\ \frac{3}{4}\left|\mathbf{Q}_{i+1} - \mathbf{Q}_i\right|, \text{ if } \left|s_1^i\right| = 1 \end{cases} \tag{10}$$

and

$$\theta_i' = -\frac{3}{4}\alpha(s_1^i - s_2^i)\min(\max(\left|\theta_1^i\right|, \left|\theta_2^i\right|), \pi - \max(\left|\theta_1^i\right|, \left|\theta_2^i\right|)). \tag{11}$$

At this point, the trajectory curve most probably has a new set of vertices, $\mathbf{Q}_1$, $\mathbf{Q}_2$, ..., $\mathbf{Q}_{n+m}$, where m is the number of new points $\mathbf{Q}_i'$ that were created. After determining the tangents at all the vertices, the trajectory curve is completely defined, and all the component curve segments $\mathbf{Q}_i\mathbf{Q}_{i+1}$ are quadratic Bézier segments.

3.2 Profile Curve

Once determined the trajectory curve, it is necessary to define the profile curve that is used to compose the basis function. The chosen profile curve has to ensure that the basis function is positive within its support domain, attains a single maximum, and does not possess local maxima or minima (see Section 2). A cubic Hermitian function guarantees all these properties and is used herein, although, other functions with similar properties could also be used.

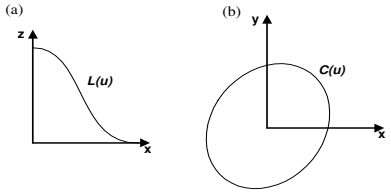

Figure 7: Configurations of Trajectory and Profile curves.

The cubic Hermitian profile curve is defined parametrically on a local coordinate system $\overline{x}\ \ \overline{y}$, where the $\overline{y}$ axis coincides with the direction of the projector $\mathbf{P}'_k\mathbf{P}_k$, by

$$\mathbf{L}(u) = [L_{\overline{x}}(u), L_{\overline{y}}(u)] = [1-u, u^2(-2u+3)], \quad 0 \leq u \leq 1. \quad (12)$$

Notice that the parameterization is such that the curve is swept from $\mathbf{L}(0) = [1, 0]$ to $\mathbf{L}(1) = [0, 1]$.

3.3 Composition of Blending Function

As mentioned at the beginning of Section 3, the blending function associated with the sample point $\mathbf{P}_k$ is generated as a swung surface by the revolution of a profile curve around the projector $\mathbf{P}'_k\mathbf{P}_k$ (its local $\overline{y}$ axis) while the profile curve is scaled by the trajectory curve during revolution (see Figure 8).

Considering, in the composition process, that the trajectory curve is defined on the plane xy as

$$\mathbf{C}(v) = [C_x(v), C_y(v)],\ 0 \leq v < 360^o, \quad (13)$$

and that, at the beginning of revolution, the profile curve is in the plane xz (Figure 7); the resulting blending function is expressed by

$$\mathbf{B}(u,v) = [L_{\overline{x}}(u)C_x(v),\ L_{\overline{x}}(u)C_y(v),\ L_{\overline{y}}(u)]. \quad (14)$$

Notice that, when $u = 0$, $\mathbf{B}(0, v) = \mathbf{C}(v)$.

The component z of the blending functions obtained, by the process just described, for the points $\mathbf{P}_k$ are the functions $w_k(x, y)$ used in Equation (2) to obtain the normalized basis functions $W_k(x, y)$.

3.4 Basis Function Contribution

The contribution of each basis function,$W_k(x, y)$, to the computation of the coordinate z, given the coordinates (x, y) of a point on the terrain's domain, is apparent in Equation (1). Therefore, considering $S = \{w_k(x, y),\ k = 1, ..., N\}$,

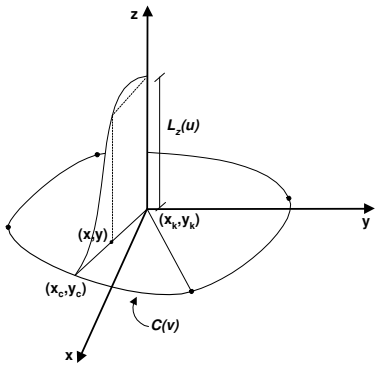

Figure 8: Contribution of basis function at the point (x, y).

where $w_k(x, y)$ is the non-normalized basis function relative to the sample point $\mathbf{P}_k$, the contribution is accomplished as follows:

1. Given a point $(x, y)\ in D$, determine the set $S_{(x,y)} \subset S$ defined as $S_{(x,y)} = \{w_l(x, y) \in S \mid w_l(x, y) > 0\}$;
2. For each basis function $w_l(x, y) \in S_{(x,y)}$, compute the intersection (x_c, y_c) of the straight-line from $\mathbf{P}'_l$ through point (x, y) with the trajectory curve corresponding to $\mathbf{P}'_l$. Scale the profile curve, i.e., compute:
$$u = 1 - \frac{(x - x_l)(x_c - x_l) + (y - y_l)(y_c - y_l)}{(x_c - x_l)^2 + (y_c - y_l)^2}, \quad (15)$$
obtaining $w_l(x, y) = L_{\overline{y}}(u)$ (Figure 8);
3. After all $w_l(x, y)$ are computed, use Equation (2) to obtain the normalized basis functions $W_l(x, y)$, and compute $S(x, y)$, e.g.
$$S(x,y) = \sum_{w_l(x,y) \in S_{(x,y)}} W_l(x,y) z_l. \quad (16)$$

3.5 Shape Control

For each chord segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$, the definition of the corresponding part of the trajectory curve involves a parameter α that determines a fraction of that chord segment, and can be used effectively for local and global shape control. In Equation (18), the choice of w determines the expansion that a Bézier segment of the trajectory curve will have relative to the corresponding chord. That expansion is limited by the requirement that the trajectory curve do not enclose points outside the triangles sharing vertex $\mathbf{P}'_k$ (see Section 3.1).

A possible way to solve this problem, if d_1 is the distance from the control point $\mathbf{b}_1$ to the chord $\mathbf{Q}_i\mathbf{Q}_{i+1}$; d_i is the distance, to this chord, from the closest non-neighboring

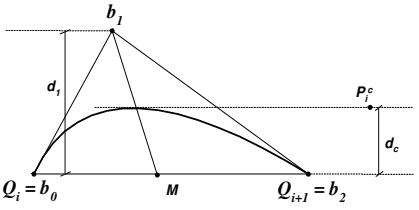

Figure 9: w for the curve segment $\mathbf{Q}_i\mathbf{Q}_{i+1}$.

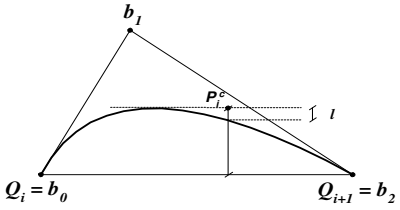

Figure 10: Bézier expansion over restricted.

point, $\mathbf{P}'_i$, in the region between the chord's support line and the parallel through point $\mathbf{b}_1$; and d_α is a fraction of that chord, given as

$$d_\alpha = \alpha \left| \mathbf{Q}_{i+1} - \mathbf{Q}_i \right|, \tag{17}$$

is given by

$$w = \min(1, (d_s)/(d_1 - d_s)), \tag{18}$$

where $d_s = \min(d_i, d_\alpha)$, if the curve is convex and $d_s = \min(d_k, d_\alpha)$, otherwise [5]. The value of w calculated by Equation (18) forces the Bézier segment to touch the line, parallel to the chord $\mathbf{Q}_i\mathbf{Q}_{i+1}$, located at a distance d_s from it [10] (see Figure 9).

One of the important properties of Bézier curves is that their expansion is limited by their convex hulls. Hence, one can conclude that, if a point $\mathbf{P}'_i$ is outside the triangle $\mathbf{b}_0\mathbf{b}_1\mathbf{b}_2$, it should not impose any restriction on the expansion of the Bézier segment (see Figure 9). Said in a different way, it is not possible for a Bézier segment to include a point $\mathbf{P}'_i$ outside its convex hull. Thus, no expansion of the Bézier curve shown in Figure 9 will be able to enclose a point $\mathbf{P}'_i$ outside the triangle $\mathbf{b}_0\mathbf{b}_1\mathbf{b}_2$. Also, it can be verified in the illustration shown in Figure 10 that, even when $\mathbf{P}'_i$ is in the interior of the convex hull $\mathbf{b}_0\mathbf{b}_1\mathbf{b}_2$, the condition that the Bézier can expand at most up to touching the straight line, parallel to the chord $\mathbf{Q}_i\mathbf{Q}_{i+1}$, passing through $\mathbf{P}'_i$ is too restrictive.

To overcome these deficiencies and to allow a more efficient expansion of the Bézier segments, a better approach is based on the following strategy for determining w:

1. Find all the sample points $\mathbf{P}'_i$ that are inside the convex hull $\mathbf{b}_0\mathbf{b}_1\mathbf{b}_2$ of the Bézier segment to be expanded;

2. For each point $\mathbf{P}'_i$ found in Step 1 and using its baricentric coordinates $({}^i\lambda_0, {}^i\lambda_1, {}^i\lambda_2)$ relative to the triangular convex hull, determine w_i by

$$w_i = ({}^i\lambda_1)/(2\sqrt{{}^i\lambda_0 \, {}^i\lambda_2}), \tag{19}$$

such that the expansion of the Bézier segment passes through the point $\mathbf{P}'_i$ [11];

3. Define $\overline{w}$ as the smallest of all w_i determined in Step 2;

4. Determine d_α according to Equation (17) and compute the final w as

$$w = \min(1, (d_\alpha)/(d_1 - d_\alpha), \overline{w}). \tag{20}$$

This choice of w takes into account just the points, not neighbors of $\mathbf{P}'_k$, that are inside the convex hull $\mathbf{b}_0\mathbf{b}_1\mathbf{b}_2$, in case they exist, and represents the largest expansion of the curve satisfying the imposed conditions. This gives more flexibility of shape control, both locally and globally, providing a more adequate fit of the interpolated terrain.

Besides the proposed shape control of the trajectory curve that allows an expansion of the basis function's support region, it can be introduced another parameter, β, that controls the shape of the profile curve $L(u)$ (Equation (12)). Thus, the modification

$$L_{\overline{y}}(u, \beta) = \beta \left[u^2(-2u + 3) \right] + (1 - \beta)u, \tag{21}$$

in the component $L_{\overline{y}}(u)$ of the function $\mathbf{L}(u)$ provides a whole range of possibilities for the profile curve. Notice that, this biased profile curve assumes any configuration between the Cubic Hermitian ($\beta = 1$) (see Equation (12)); and a linear function ($\beta = 0$). The tangents at $u = 0$ and $u = 1$ are both equal to $(1 - \beta)$. Therefore, at the sample points, unless $\beta = 1$, the digital terrain model does not have to show a plateau shape. In Figure 11, some examples of the same basis function with different α and β are shown.

4 Comparison of Results

In this section, two reference models were chosen to show the comparison tests. Each reference model is compared to three interpolated ones generated by: polyhedral interpolation, the interpolation method proposed in [5], and the interpolation method proposed in this work.

These three interpolated models were constructed as follows: 1) a number of sample points was defined on the domain; 2) a Delaunay triangulation of these points was determined; 3) the elevations for these points were directly

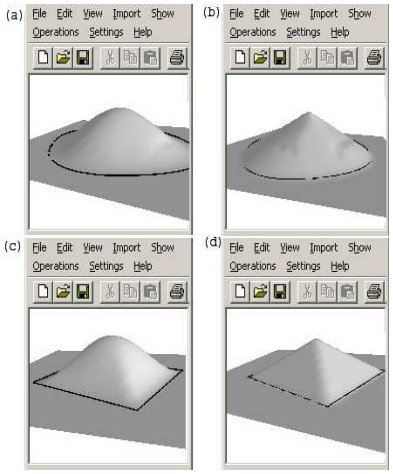

Figure 11: Basis function with different α and β. (a) $\alpha = \beta = 1$ (b) $\alpha = 1$ and $\beta = 0$, (c) $\alpha = 0$ and $\beta = 1$ and (d) $\alpha = \beta = 0$.

determined from the reference model; 4) the interpolated surfaces corresponding to each method were computed.

With appropriate values of the parameters α and β (see Section 3.5), particular cases of interpolation can be obtained. For instance, the polyhedral interpolation is defined with values of α and β equal to zero. Thus, for comparison with the reference surface, interpolated surfaces were determined by each of the three methods, using α and β equal to 0.25, 0.50, 0.75 and 1.0. The fact that α in the proposed method allows a larger expansion of the trajectory curves gives more flexibility to model more complex surfaces.

The errors ϵ between the interpolated surfaces and the reference one were calculated using the expression

$$\varepsilon = \frac{\sqrt{\Sigma_{i,j}\left(H(x_i, y_j) - S(x_i, y_j)\right)^2}}{\sqrt{\Sigma_{i,j} H(x_i, y_j)^2}}, \qquad (22)$$

where (x_i, y_j) are the points of the regular grid, $H(x_i, y_j)$ represents the elevation of the reference surface at (x_i, y_j), and $S(x_i, y_j)$ represents the elevation of the interpolated surface determined at (x_i, y_j). The purpose of this study is only to evaluate the potential of the proposed method, in a simple way, without focusing on the magnitude of the errors found. In other words, the idea is to show that smaller errors can be found with the proposed method, which shows a tendency that the method can generate better interpolated surfaces in real applications. This is probably more true in applications with steep regions and slopes, although further investigation is necessary.

	sinc function (200 points)		
	Polyhedral	Chaturvedi	Proposed
$\alpha = \beta = 0.25$	40.35	39.71	37.78
$\alpha = \beta = 0.50$	40.35	40.08	37.46
$\alpha = \beta = 0.75$	40.35	40.74	37.29
$\alpha = \beta = 1.00$	40.35	41.63	37.31

Table 1: Comparison of the percentage errors.

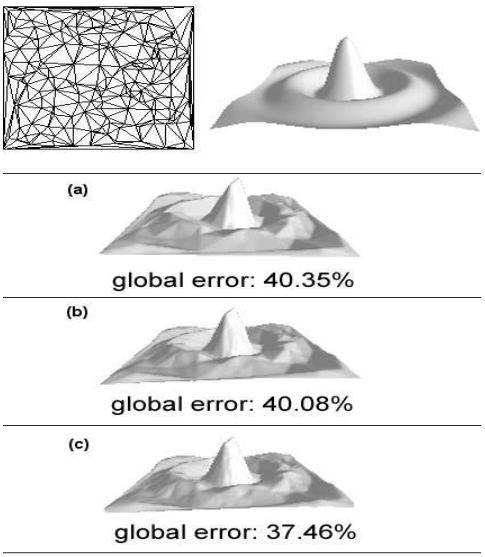

Figure 12: Comparison of the interpolation methods ($\alpha = 0.5$ and $\beta = 0.5$): (a) Polyhedral, (b) Chaturvedi (c) Proposed. (200 sample points)

4.1 Case 1: *Sinc* function

In this example, the reference model was determined as the *Sinc* function

$$F(x, y) = (\sin(\sqrt{x^2 + y^2}))/(\sqrt{x^2 + y^2}), \qquad (23)$$

defined on the domain $D([-12, 12] \times [-12, 12]) \subset R^2$. The interpolated models were constructed as follows: 1) 200 random sample points were considered in the domain D; 2) a Delaunay triangulation of these points was established; 3) their elevations computed by $z_i = F(x_i, y_i)$; 4) the interpolated surfaces were determined.

The three interpolated surfaces are shown in Figure 12. For each case, the approximation error was calculated by Equation (22), as shown in Table 1.

4.2 Case 2: Real terrain

In this example, the reference model was defined on the domain $D([-12, 12] \times [-12, 12]) \subset R^2$. This model was defined by 10.000 points, distributed as a regular grid of 100×100 points over the entire domain D. This example was chosen to illustrate the accuracy of the proposed

	Real model (57 points)		
	Polyhedral	Chaturvedi	Proposed
$\alpha = \beta = 0.25$	19.98	18.67	18.27
$\alpha = \beta = 0.50$	19.98	18.40	17.40
$\alpha = \beta = 0.75$	19.98	18.44	16.83
$\alpha = \beta = 1.00$	19.98	18.84	16.73

Table 2: Comparison of the percentage errors.

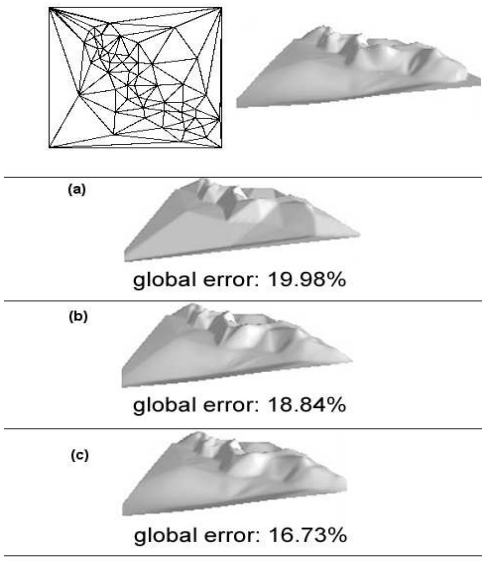

Figure 13: Comparison of interpolation methods ($\alpha = \beta = 1$: (a) Polyhedral, (b) Chaturvedi and (c) Proposed.

technique applied to a real terrain. The digital terrain models (interpolated models) were constructed as follows: 1) 57 sample points were chosen from the set of points that defined the reference model (part of the points was chosen at random and some points were manually chosen on steep regions); 2) a Delaunay triangulation of these points was established; 3) their elevations were obtained from the reference model; 4) the interpolated surfaces were determined. The number of sample points chosen was intentionally small in order to try to represent the real model with the minimum computational effort possible, without losing too much accuracy.

The three interpolated surfaces are shown in Figure 13. For each case, the approximation error was calculated by Equation (22), as shown in Table 2.

5 Conclusions

In this paper, it was proposed a methodology for generation of interpolation surfaces by modifying the basis function generation process presented by Chaturvedi and Piegl [5]. The proposed modification allows for a larger expansion of the basis function's support region, which is represented by the interior of a trajectory curve composed of quadratic rational Bézier segments. It also reduces the approximation error between the reference and the interpolation surfaces.

It was observed that, in all examples, the proposed method presented a smaller approximation error than both the polyhedral method and that of Chaturvedi and Piegl. Although further investigation needs to be conducted, this fact seems to be very promising.

References

[1] Heckbert, P. S. and Garland, M. *Fast Polygonal Approximation of Terrain and Height Fields.* Tech. report CMU-CS-95-181, CS Department, Carnegie Mellon University, September, 1995.

[2] Amenta, N.; Bern, M. and Kamvysselis, M. *A New Voronoi-Based Surface Reconstruction Algorithm.* In SIGGRAPH '98 Conf. Proc., pp. 415–422, 1998.

[3] Bajaj, C.; Bernardini, F. and Xu, G. *Reconstructing Surfaces and Functions on Surfaces from Unorganized 3D Data.* Algorithmica, 19, pp. 243–261, 1997.

[4] Lodha, S. K. and Franke, R. *Scattered Data Techniques for Surfaces*. In Proc. of Dagstuhl Conf. on Scientific Visualization, (eds.) H. Hagen, G. Nielson and F. Post, IEEE Comp. Soc. Press, pp. 182–222, 1999.

[5] Chaturvedi, A. K. and Piegl L. A. *Procedural Method for Terrain Surface Interpolation.* In Comp.& Graphics, 20(4), pp. 541–566, 1996.

[6] Bajaj, C. L. *Surface Fitting with Implicit Algebraic Surfaces Patches.* In Topics in Surf. Modeling, pp. 23–52, SIAM, Philadelphia, 1992.

[7] Mann, S.; Loop, C.; Lounsbery, M.; Meyers, D.; Painter, J.; DeRose T. and Sloan, K. *A Survey of Parametric Scattered Data Fitting Using Triangular Interpolants*. In Curve and Surf. Design, H. Hagen (ed.), pp. 145–172, SIAM, 1992.

[8] Nielson, G. M. *Scattered Data Modeling.* IEEE Comp. Graphics and Applic., 13, pp. 60–70, 1993.

[9] Piegl, L. A. and Tiller, W. *The NURBS book.* Springer-Verlag, 1995.

[10] Ahn, Y. J. and Qim, H. O. *Curvatures of the Quadratic Rational Bezier Curves.* In Comp. & Mathematics Applic., 36(9), Elsevier Science Ltd., 1998.

[11] Fudos, I. and Hoffmann, C. M. *Constraint-Based Parametric Conics for CAD.* In Comp. Aided Design 28(2), pp. 91–100, 1996.