Received: 30 November 2021 | Revised: 18 April 2022 | Accepted: 13 July 2022

DOI: 10.1002/mma.8602

RESEARCH ARTICLE

WILEY

Spline quasi-interpolation in the Bernstein basis and its application to digital elevation models

Francisco J. Ariza-López[1] | **Domingo Barrera**[2] | **Salah Eddargani**[2] | **María José Ibáñez**[2] | **Juan F. Reinoso**[3]

[1]Department of Cartographic, Geodesic and Photogrammetry Engineering, University of Jaén, Jaén, Spain

[2]Department of Applied Mathematics, University of Granada, Granada, Spain

[3]Department of Architectural and Engineering Graphic Expression, University of Granada, Granada, Spain

Correspondence
Salah Eddargani, Department of Applied Mathematics, University of Granada, Granada, Spain.
Email: seddargani@correo.ugr.es

Communicated by: J. Vigo-Aguiar

Funding information
Spanish State Research Agency, Grant/Award Number: PID2019-106195RB-I00; Universidad de Granada/CBUA

A nonstandard low-cost spline approximation method for approximating bivariate functions is constructed. It is applied for Digital Elevation approximation and then its accuracy in the downscaling process is studied.

KEYWORDS

altimetric error, Bernstein–Bézier coefficients, DEM, upscaling, downscaling, tensor product, quasi-interpolation

1 | INTRODUCTION

Digital Terrain Elevation Models (DETM) are cartographic products with multiple applications in fields such as civil engineering, hydrology, agriculture, environment, and geology.[1] The quality of the results achieved in each field will largely depend on the data capture method and the transformation of the captured data into a type of DETM. The two most important capture methods currently are photogrammetry and light detection and ranging (LiDAR) for their ability to cover large areas of land that characterizes cartography. To achieve these large territorial areas, data capture is carried out with airborne sensors. The result of the data capture is a point cloud that after processing can be transformed into two types of DETM: An irregular triangle network (TIN) or a regular mesh (DEM). The TIN would be defined by a point number and its corresponding 3D coordinates together with the relationship of adjacent points, which it forms edges of the corresponding triangles with. On the other hand, the DEM would be defined by a matrix in which the value of each cell corresponds to the altitude and the planimetric coordinates are easily calculated knowing the cell size in addition to the coordinates of the first row and column of the matrix.

We will focus on DEMs as one of the products delivered by most of the national and regional cartographic agencies of the states. DEMs, although they represent a territorial area, have a discrete character, typical of the matrices through which they are represented. Thus, if it is desired to obtain the slope of the ground, it could be only known in the coordinates associated with the matrix of cell centers. Other variables derived from the DEM could be the orientation, curvature, or flow accumulation, which determines the drainage network, and its computation presents the same problem of spatial uncertainty as the above-mentioned slope.

The discontinuity of DEMs poses a problem when it is necessary to compare variables derived from two DEMs that represent the same territorial space, but each one having a different cell size. Upscaling[2] and downscaling[3–6] are commonly used techniques with regard to this problem, and different approximation methods have been employed for this purpose, such as nearest neighbor, bilinear[7] or bicubic[8] interpolation.

In this paper, we propose a novel method of function approximation that will allow us to perform both upscaling and downscaling, and we will study its accuracy in the downscaling process. For the function underlying a DEM, we will construct a nonstandard quasi-interpolant that will provide a C^2 bicubic piecewise surface from which it will be possible to estimate each of the elements required in practice. Quasi-interpolation is a well-known technique for constructing approximants directly from the available information of the function to be approximated, which can be reduced to the values it takes on a set of points. Typically, a quasi-interpolant for a given function will be a linear combination of elements of a set of appropriate non-negative and compactly supported functions, which form a partition of unity. It can be defined to satisfy some required properties.[9–13] The nonstandard feature of the quasi-interpolant constructed here comes from the fact that it will be a tensor product quasi-interpolant defined from a univariate non-standard quasi-interpolant, which is constructed on each sub-interval by providing the coefficients in the Bernstein basis. Therefore, on each square of the quadrangulation associated with the DEM the coefficients of the representation of the quasi-interpolant in terms of the corresponding Bernstein polynomials are directly defined from the values at the points in a neighborhood.[14–17]

In Section 2 a nonstandard univariate quasi-interpolant is constructed for functions defined on the real line, endowed with a uniform partition. Some numerical tests are presented to show the performance of this kind of spline quasi-interpolant. Next, it is appropriately modified to approximate functions defined over an interval. The nonstandard bivariate quasi-interpolant is obtained as the tensor product of the univariate scheme with itself. Numerical tests are presented.

In Section 3 the approximation method proposed is applied to a DEM and the altimetric error will be analyzed by comparing altitudes and the planimetric error using the automatic contour line algorithm.[18,19]

2 | QUASI-INTERPOLATION IN THE BERNSTEIN BASIS

2.1 | The univariate case

Suppose that for a real function f the values $f(v_i)$ and $f(e_i)$, $i \in \mathbb{Z}$, are known, where $v_i := a + ih$ and $e_i := v_i + \frac{1}{2}h$, with $h > 0$ the size of the partition $\Delta := \{v_i, i \in \mathbb{Z}\}$, and $a \in \mathbb{R}$.

We want to construct a cubic spline Qf of f defined on Δ. Qf reduces on each interval $I_i := [v_i, v_{i+1}]$, $i \in \mathbb{Z}$, to a cubic polynomial, so that it can be expressed in Bernstein's basis relative to I_i, that is,

$$Qf_{|I_i}(x) = \sum_{k=0}^{3} b_{k,i} B_k\left(\frac{x - v_i}{h}\right), x \in I_i, \tag{1}$$

for some coefficients $b_{k,i} \in \mathbb{R}$, where

$$B_k(t) := \binom{3}{k} t^k (1-t)^{3-k}, t \in [0,1],$$

are the cubic Bernstein polynomials relative to the interval $[0,1]$. The Bernstein–Bézier (BB) coefficients $b_{k,i}$ of Qf on I_i are linked to the domain points $v_i + \frac{k}{3}h$, $k = 0, \ldots, 3$. The union without repetition of all of them gives rise to the set $\mathcal{D} := \left\{a + i\frac{h}{3}, i \in \mathbb{Z}\right\}$.

The BB-coefficients of the quasi-interpolant will be determined to get C^2 continuity and exactness on the space $\mathbb{P}_2$ of quadratic polynomials. Since the partition is uniform, one strategy is to start from a partition of $\mathcal{D}$ by subsets $\mathcal{D}_i$ and appropriately determine the BB-coefficients corresponding to the points in $\mathcal{D}_i$. Specifically, it is fulfilled that $\mathcal{D} = \bigcup_{i\in\mathbb{Z}} \mathcal{D}_i$, with $\mathcal{D}_i := \{u_i, v_i, w_i\}$, where $u_i := v_i - \frac{h}{3}$ and $w_i := v_i + \frac{h}{3}$. In Figure 1, the above construction is illustrated. The upper subfigure illustrates the structure of the set $\mathcal{D}$. In the lower subfigure, the subsets $\mathcal{D}_i$ corresponding to various vertices are shown.

According to the notation introduced above, the quasi-interpolant defined in (1) is rewritten as

$$Qf_{|I_i}(x) = c(v_i) B_0\left(\frac{x - v_i}{h}\right) + c(w_i) B_1\left(\frac{x - v_i}{h}\right) + c(u_{i+1}) B_2\left(\frac{x - v_i}{h}\right) + c(v_{i+1}) B_3\left(\frac{x - v_i}{h}\right), x \in I_i, \tag{2}$$

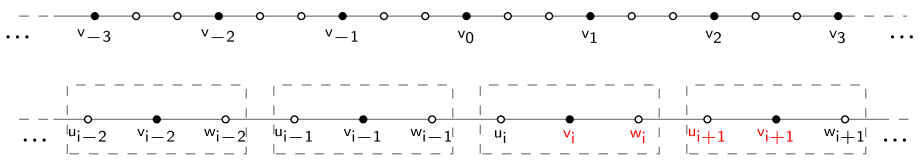

FIGURE 1 (Top) Subset of domain points in D for some subintervals induced by Δ. (Bottom) Labeling domain points of various subintervals [Colour figure can be viewed at wileyonlinelibrary.com]

where $c(p)$ stands for the BB-coefficient of the domain point p. Figure 1 shows the four domain points whose associated BB-coefficients are involved in the above expression.

We propose to define the BB-coefficients of Qf in each interval I_i shown in (2) as linear combinations of the values of f at knots v_ℓ and midpoints e_ℓ in a small neighborhood of that interval.

Problem 1. Find masks $\alpha := (\alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4)$, $\beta := (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4)$ and $\gamma := (\gamma_0, \gamma_1, \gamma_2, \gamma_3, \gamma_4)$ such that the quasi-interpolant Qf defined by (2) on each interval I_ℓ with BB-coefficients

$$\begin{aligned} c(u_\ell) &:= \alpha_0 f(v_{\ell-1}) + \alpha_1 f(e_{\ell-1}) + \alpha_2 f(v_\ell) + \alpha_3 f(e_\ell) + \alpha_4 f(v_{\ell+1}), \\ c(v_\ell) &:= \beta_0 f(v_{\ell-1}) + \beta_1 f(e_{\ell-1}) + \beta_2 f(v_\ell) + \beta_3 f(e_\ell) + \beta_4 f(v_{\ell+1}), \\ c(w_\ell) &:= \gamma_0 f(v_{\ell-1}) + \gamma_1 f(e_{\ell-1}) + \gamma_2 f(v_\ell) + \gamma_3 f(e_\ell) + \gamma_4 f(v_{\ell+1}) \end{aligned} \tag{3}$$

is C^2 continuous and $Qf = f$ for all $f \in \mathbb{P}_2$.

The fact that the partition is uniform leads to the imposition that the masks do not depend on the subinterval in which the quasi-interpolant is calculated.

Proposition 2. *The quasi-interpolant Qf defined locally by (2) and (3) is C^1-continuous if and only if*

$$\alpha_k - 2\beta_k + \gamma_k = 0, k \in \{0, 1, 2, 3, 4\}. \tag{4}$$

Proof. Qf is C^1-continuous if and only if $c(w_\ell) - 2c(v_\ell) + c(u_\ell) = 0$ for all $\ell \in \mathbb{Z}$. Replacing the BB-coefficients $c(w_\ell)$, $c(v_\ell)$, and $c(u_\ell)$ with their expressions given in (3), it holds

$$c(w_\ell) - 2c(v_\ell) + c(u_\ell) = E_{0,1} f(v_{\ell-1}) + E_{1,1} f(e_{\ell-1}) + E_{2,1} f(v_\ell) + E_{3,1} f(e_\ell) + E_{4,1} f(v_{\ell+1}),$$

where $E_{k,1} := \alpha_k - 2\beta_k + \gamma_k$. This expression will be zero for any function f if and only if the conditions in the statement are satisfied. □

Proposition 3. *Let us suppose that the quasi-interpolant Qf defined locally by (2) and (3) is C^1-regular. Then, it is C^2-continuous if and only if*

$$2\alpha_0 - 2\gamma_0 - \gamma_2 = 0,\ 2\alpha_1 - 2\gamma_1 - \gamma_3 = 0,\ \alpha_0 + 2\alpha_2 - 2\gamma_2 - \gamma_4 = 0,\ \alpha_1 + 2\alpha_3 - 2\gamma_3 = 0,\ \alpha_2 + 2\alpha_4 - 2\gamma_4 = 0. \tag{5}$$

Proof. Being Qf C^1-continuous, C^2-continuity is achieved if and only if $c(u_{\ell+1}) + 2c(u_\ell) - 2c(w_\ell) - c(w_{\ell-1}) = 0$ for all $\ell \in \mathbb{Z}$. Replacing $c(u_{\ell+1})$, $c(u_\ell)$, $c(w_\ell)$, and $c(w_{\ell-1})$ with their expressions given in (3), and taking into account that β_k depends on α_k and γ_k, then

$$c(u_{\ell+1}) + 2c(u_\ell) - 2c(w_\ell) - c(w_{\ell-1}) = E_{0,2} f(v_{\ell-1}) + E_{1,2} f(e_{\ell-1}) + E_{2,2} f(v_\ell) + E_{3,2} f(e_\ell) + E_{4,2} f(v_{\ell+1}),$$

where $E_{0,2} := 2\alpha_0 - 2\gamma_0 - \gamma_2$, $E_{1,2} := 2\alpha_1 - 2\gamma_1 - \gamma_3$, $E_{2,2} := \alpha_0 + 2\alpha_2 - 2\gamma_2 - \gamma_4$, $E_{3,2} := \alpha_1 + 2\alpha_3 - 2\gamma_3$ and $E_{4,2} := \alpha_2 + 2\alpha_4 - 2\gamma_4$. Therefore, Qf is C^2-continuous if and only if $E_{k,2} = 0$, $k = 0, \dots, 4$, and the proof is complete. □

The following result is easily obtained.

Lemma 4. *The BB-coefficients in the interval I_ℓ of monomials $m_k(x) := \left(\frac{x - v_i}{h}\right)^k$, $k = 0, 1, 2$, are $(1, 1, 1, 1)$, $(0, 1/3, 2/3, 1)$ and $(0, 0, 1/3, 1)$, respectively.*

We are now in a position to state the main result of this sub-section.

Proposition 5. *Problem 1 has a unique solution, provided by the masks* $\alpha = \left(0, \frac{8}{15}, \frac{2}{5}, \frac{4}{15}, -\frac{1}{5}\right)$, $\beta = \left(-\frac{1}{10}, \frac{2}{5}, \frac{2}{5}, \frac{2}{5}, -\frac{1}{10}\right)$ *and* $\gamma = \left(-\frac{1}{5}, \frac{4}{15}, \frac{2}{5}, \frac{8}{15}, 0\right)$.

Proof. By (2), the BB-coefficients relative to interval I_ℓ of the quasi-interpolant Qf are $(c(v_\ell), c(w_\ell), c(u_{\ell+1}), c(v_{\ell+1}))$ and are given by (3). The BB-coefficients satisfy conditions (4) and (5) to ensure C^2-continuity of Qf. Moreover, exactness on $\mathbb{P}_2$ is required, so that BB-coefficients of Qm_k on I_ℓ must be equal to those of m_k. They are given in Lemma 4 and those of Qm_k are $(1,1,1,1)$, $\left(\frac{1}{2}(-2\gamma_0 - \gamma_1 + \gamma_3 + 2\gamma_4), \frac{1}{2}(-2\beta_0 - \beta_1 + \beta_3 + 2\beta_4), \frac{1}{2}(\alpha_1 + 2\alpha_2 + 3\alpha_3 + 4\alpha_4), \frac{1}{2}(\gamma_1 + 2\gamma_1 + 3\gamma_3 + 4\gamma_4)\right)$ and

$$\left(\frac{1}{4}(4\gamma_0 + \gamma_1 + \gamma_3 + 4\gamma_4), \frac{1}{4}(4\beta_0 + \beta_1 + \beta_3 + 4\beta_4), \frac{1}{4}(\alpha_1 + 4\alpha_2 + 9\alpha_3 + 16\alpha_4), \frac{1}{4}(\gamma_0 + 4\gamma_1 + 9\gamma_3 + 16\gamma_4)\right).$$

Therefore, there are 10 linear equations that guarantee C^2-continuity and 12 other that produce the required exactness. In total, there are 22 equations in 15 unknowns. A symbolic calculus system makes it possible to prove that such a system of equations has a unique solution, which gives rise to the masks indicated in the statement. □

For $f \in C^3(I_\ell)$ and for the quasi-interpolation operator Q, given by the unique solution to Problem 1, there exists a constant C_ℓ independent of h such that

$$\|f - Qf\|_{\infty, I_\ell} \leq C_\ell h^3 \left\|f^{(3)}\right\|_{\infty, I_\ell}.$$

2.2 | Numerical tests in the univariate case

To test the performance of the quasi-interpolation scheme defined above, we consider the following three test functions, defined on $[0, 1]$, although they can be extended to a larger interval:

1. $f_1(x) := \frac{3}{4}e^{-2(9x-2)^2} - \frac{1}{5}e^{-(9x-7)^2-(9x-4)^2} + \frac{1}{2}e^{-(9x-7)^2-\frac{1}{4}(9x-3)^2} + \frac{3}{4}e^{\frac{1}{10}(-9x-1)-\frac{1}{49}(9x+1)^2}$.
2. $f_2(x) := \frac{1}{2}x\cos^4\left(4\left(x^2 + x - 1\right)\right)$.
3. $f_3(x) := (3 + \cos(2\pi x))^{-1} \log\left(\frac{1}{x^3+1}\right) \sin(7\pi x)$.

Their plots appear in Figure 2.

The tests are carried out for a sequence of uniform partition Δ_n associated with the vertices $v_i = ih$, $i = 0, \ldots, n$, where $h := \frac{1}{n}$.

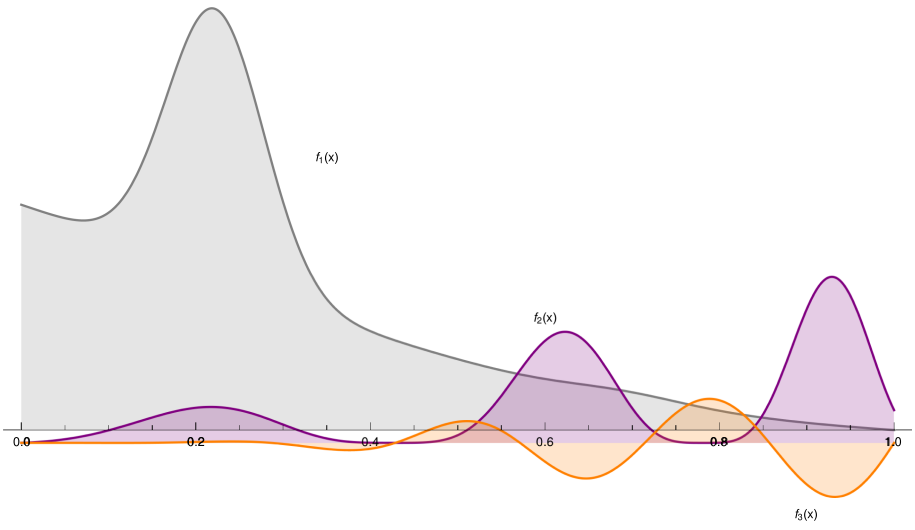

FIGURE 2 Plots of univariate functions f_1, f_2, and f_3 [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Error estimates and NCOs for test functions f_1, f_2, and f_3

	f_1		f_2		f_3	
n	Error	NCO	Error	NCO	Error	NCO
16	0.04662822716	–	0.03579025054	–	0.008119172533	–
32	0.004072919573	3.517	0.005506927133	2.700	0.0007660623721	3.406
64	0.000327887317	3.635	0.0004341112452	3.665	0.00007860374495	3.285
128	0.000033860570	3.276	0.00004277811450	3.343	$8.9665358890173 \times 10^{-6}$	3.132
256	$4.072326029745 \times 10^{-6}$	3.056	$5.5514176444525 \times 10^{-6}$	2.946	$1.0827003465402 \times 10^{-6}$	3.050

The quasi-interpolation error is estimated as

$$\max_{\ell=1,\dots,200} |f(x_\ell) - Qf(x_\ell)|,$$

where x_ℓ, $\ell = 1, \dots, 200$, are equally spaced points in $[0,1]$. The numerical convergence order (NCO) is given by the rate

$$\text{NCO} := \log_2\left(\frac{E(2n)}{E(n)}\right),$$

where $E(m)$ stands for the estimated error associated with Δ_m.

Table 1 shows the numerical errors and the numerical convergence orders (NCOs) for f_1, f_2, and f_3.

2.3 | Quasi-interpolation of functions defined on an interval

When the function f is defined on an interval $[a,b]$ endowed with a uniform partition with knots $x_i := a + ih$, $i = 0, \dots, n$, $h := \frac{b-a}{n}$, then the above numerical scheme is not applicable since the values $f(a-h)$, $f\left(a-\frac{h}{2}\right)$, $f\left(b+\frac{h}{2}\right)$ and $f(b+h)$, which should appear in (3), are not defined. Next, it is shown how to define those values.

Lemma 6. *Let us suppose that $f(a-h)$ and $f\left(a-\frac{h}{2}\right)$ are approximated by $\phi_0 f(a) + \phi_1 f\left(a+\frac{h}{2}\right) + \phi_2 f(a+h)$ and $\varphi_0 f(a) + \varphi_1 f\left(a+\frac{h}{2}\right) + \varphi_2 f(a+h)$. Then, the approximation errors are null when the data come from a quadratic polynomial if and only if $\phi_0 = 6$, $\phi_1 = -8$, $\phi_2 = 3$, $\varphi_0 = 3$, $\varphi_1 = -3$, and $\varphi_2 = 1$. Similarly, if $f\left(b+\frac{h}{2}\right)$ and $f(b+h)$ are approximated by $\overline{\varphi}_0 f(b) + \overline{\varphi}_1 f\left(b-\frac{h}{2}\right) + \overline{\varphi}_2 f(b-h)$ and $\overline{\phi}_0 f(b) + \overline{\phi}_1 f\left(b-\frac{h}{2}\right) + \overline{\phi}_2 f(b-h)$, then the approximation errors satisfy the same condition regarding exactness if and only if $\overline{\varphi}_0 = 1$, $\overline{\varphi}_1 = -3$, $\overline{\varphi}_2 = 3$, $\overline{\phi}_0 = 3$, $\overline{\phi}_1 = -8$, and $\overline{\phi}_2 = 6$.*

Proof. The results are obtained by imposing that the errors are zero when $f = 1, x, x^2$, and solving the resulting systems of equations. □

Substituting in the BB-coefficients corresponding to $\ell = 0$ and $\ell = n-1$ given by (3) and in Proposition 5 the approximations obtained to $f(a-h)$, $f\left(a-\frac{h}{2}\right)$, $f\left(b+\frac{h}{2}\right)$ and $f(b+h)$, we obtain the BB-coefficients in the subintervals $[a, a+h]$ and $[b-h, b]$. They are given next.

Proposition 6. *The BB-coefficients of the quasi-interpolant Qf to $f \in C([a,b])$ in $[a, a+h]$ are*

$$\begin{aligned}
&f(a),\\
&\frac{4}{3} f\left(a+\frac{h}{2}\right) - \frac{1}{3} f(a+h),\\
&\frac{8}{15} f\left(a+\frac{h}{2}\right) + \frac{2}{5} f(a+h) + \frac{4}{15} f\left(a+\frac{3}{2}h\right) - \frac{1}{5} f(a+2h),\\
&-\frac{1}{10} f(a) + \frac{2}{5} f\left(a+\frac{h}{2}\right) + \frac{2}{5} f(a+h) + \frac{2}{5} f\left(a+\frac{3}{2}h\right) - \frac{1}{10} f(a+2h).
\end{aligned}$$

Those in $[b-h, b]$ *are*

$$
\begin{aligned}
&-\frac{1}{10}f(b-2h)+\frac{2}{5}f\left(b-\frac{3}{2}h\right)+\frac{2}{5}f(b-h)+\frac{2}{5}f\left(b-\frac{h}{2}\right)-\frac{1}{10}f(b),\\
&-\frac{1}{5}f(b-2h)+\frac{4}{15}f\left(b-\frac{3}{2}h\right)+\frac{2}{5}f(b-h)+\frac{8}{15}f\left(b-\frac{h}{2}\right),\\
&-\frac{1}{3}f(b-h)+\frac{4}{3}f\left(b-\frac{h}{2}\right),\\
&f(b).
\end{aligned}
$$

2.4 | Nonstandard quasi-interpolation of bivariate functions

The aim of this subsection is to define a quasi-interpolant for a bivariate function $f(x, y)$ defined on a square $[a, b] \times [a, b]$ from the nonstandard quasi-interpolation operator Q constructed from the *general masks* in Proposition 5 and the *boundary masks* in Proposition 6. Low computational cost is an essential feature of the bivariate numerical approximation scheme to be used in modeling DEMs which, in general, will correspond to large-area terrains, so that Q will perform well a priori.

The 2D quasi-interpolant is defined as the tensor product of Q with itself. For $y \in [a, b]$ and $x \in I_i$, $i = 1, \ldots, n-2$, the quasi-interpolant $Qf(\cdot, y)$ of f as a function of the first variable is written as

$$
\begin{aligned}
g(y) := {} & B_{i,0}(x)\left(\beta_0 f(v_{i-1}, y)+\beta_1 f(e_{i-1}, y)+\beta_2 f(v_i, y)+\beta_3 f(e_i, y)+\beta_4 f(v_{i+1}, y)\right)\\
& + B_{i,1}(x)\left(\gamma_0 f(v_{i-1}, y)+\gamma_1 f(e_{i-1}, y)+\gamma_2 f(v_i, y)+\gamma_3 f(e_i, y)+\gamma_4 f(v_{i+1}, y)\right)\\
& + B_{i,2}(x)\left(\alpha_0 f(v_i, y)+\alpha_1 f(e_i, y)+\alpha_2 f(v_{i+1}, y)+\alpha_3 f(e_{i+1}, y)+\alpha_4 f(v_{i+2}, y)\right)\\
& + B_{i,3}(x)\left(\beta_0 f(v_i, y)+\beta_1 f(e_i, y)+\beta_2 f(v_{i+1}, y)+\beta_3 f(e_{i+1}, y)+\beta_4 f(v_{i+2}, y)\right),
\end{aligned}
$$

where $B_{i,k}(x) := B_k\left(\frac{x-v_i}{h}\right)$, $k = 0, 1, 2, 3$. Therefore, for $j = 1, \ldots, n-2$, the tensor product quasi-interpolant $T_{i,j}f$ of $f(x, y)$ on $I_i \times I_j$ is given by

$$
\begin{aligned}
T_{i,j}f(x, y) := {} & B_{j,0}(y)\left(\beta_0 g(v_{j-1})+\beta_1 g(e_{j-1})+\beta_2 g(v_j)+\beta_3 g(e_j)+\beta_4 g(v_{j+1})\right)\\
& + B_{j,1}(y)\left(\gamma_0 g(v_{j-1})+\gamma_1 g(e_{j-1})+\gamma_2 g(v_j)+\gamma_3 g(e_j)+\gamma_4 g(v_{j+1})\right)\\
& + B_{j,2}(y)\left(\alpha_0 g(v_j)+\alpha_1 g(e_j)+\alpha_2 g(v_{j+1})+\alpha_3 g(e_{j+1})+\alpha_4 g(v_{j+2})\right)\\
& + B_{j,3}(y)\left(\beta_0 g(v_j)+\beta_1 g(e_j)+\beta_2 g(v_{j+1})+\beta_3 g(e_{j+1})+\beta_4 g(v_{j+2})\right),
\end{aligned}
$$

with $B_{j,k}(y) := B_k\left(\frac{y-v_j}{h}\right)$, $k = 0, 1, 2, 3$.

After some calculations, $T_{i,j}$ is found to be a linear combination of the Bernstein polynomials associated with the square $I_i \times I_j$, whose coefficients are determined from the values of f at vertices and midpoints of a neighborhood of that square. Figure 3 shows how the Bernstein polynomials and the corresponding coefficients are arranged in a rectangular structure.

$B_{i,0}B_{j,3}$	$B_{i,1}B_{j,3}$	$B_{i,2}B_{j,3}$	$B_{i,3}B_{j,3}$	$\phi_{0,3}$	$\phi_{1,3}$	$\phi_{2,3}$	$\phi_{3,3}$
$B_{i,0}B_{j,2}$	$B_{i,1}B_{j,2}$	$B_{i,2}B_{j,2}$	$B_{i,3}B_{j,2}$	$\phi_{0,2}$	$\phi_{1,2}$	$\phi_{2,2}$	$\phi_{3,2}$
$B_{i,0}B_{j,1}$	$B_{i,1}B_{j,1}$	$B_{i,2}B_{j,1}$	$B_{i,3}B_{j,1}$	$\phi_{0,1}$	$\phi_{1,1}$	$\phi_{2,1}$	$\phi_{3,1}$
$B_{i,0}B_{j,0}$	$B_{i,1}B_{j,0}$	$B_{i,2}B_{j,0}$	$B_{i,3}B_{j,0}$	$\phi_{0,0}$	$\phi_{1,0}$	$\phi_{2,0}$	$\phi_{3,0}$

FIGURE 3 Arrangement of Bernstein polynomials relative to square $I_i \times I_j$ and its associated BB-coefficients

More precisely,

$$T_{i,j}f(x,y)=\sum_{p=0}^{3}\sum_{q=0}^{3}\phi_{p,q}B_p\left(\frac{x-v_i}{h}\right)B_q\left(\frac{y-v_j}{h}\right),\ (x,y)\in I_i\times I_j,\ i,j\in\{1,\ldots,n-2\},$$

with

$$\begin{aligned}
\phi_{0,0}=&\beta_0^2 f\left(v_{i-1},v_{j-1}\right)+\beta_0\beta_1\left(f\left(e_{i-1},v_{j-1}\right)+f\left(v_{i-1},e_{j-1}\right)\right)+\beta_1^2 f\left(e_{i-1},e_{j-1}\right)+\beta_0\beta_2\left(f\left(v_{i-1},v_j\right)+f\left(v_i,v_{j-1}\right)\right)\\
&+\beta_1\beta_2\left(f\left(e_{i-1},v_j\right)+f\left(v_i,e_{j-1}\right)\right)+\beta_2^2 f\left(v_i,v_j\right)+\beta_0\beta_3\left(f\left(e_i,v_{j-1}\right)+f\left(v_{i-1},e_j\right)\right)\\
&+\beta_1\beta_3\left(f\left(e_{i-1},e_j\right)+f\left(e_i,e_{j-1}\right)\right)+\beta_2\beta_3\left(f\left(e_i,v_j\right)+f\left(v_i,e_j\right)\right)+\beta_3^2 f\left(e_i,e_j\right)+\beta_0\beta_4\left(f\left(v_{i-1},v_{j+1}\right)+f\left(v_{i+1},v_{j-1}\right)\right)\\
&+\beta_1\beta_4\left(f\left(e_{i-1},v_{j+1}\right)+f\left(v_{i+1},e_{j-1}\right)\right)+\beta_2\beta_4\left(f\left(v_i,v_{j+1}\right)+f\left(v_{i+1},v_j\right)\right)+\beta_3\beta_4\left(f\left(e_i,v_{j+1}\right)+f\left(v_{i+1},e_j\right)\right)\\
&+\beta_4^2 f\left(v_{i+1},v_{j+1}\right),\\
\phi_{0,1}=&\beta_0\gamma_0 f\left(v_{i-1},v_{j-1}\right)+\beta_1\gamma_0 f\left(e_{i-1},v_{j-1}\right)+\beta_2\gamma_0 f\left(v_i,v_{j-1}\right)+\beta_3\gamma_0 f\left(e_i,v_{j-1}\right)+\beta_4\gamma_0 f\left(v_{i+1},v_{j-1}\right)+\beta_0\gamma_1 f\left(v_{i-1},e_{j-1}\right)\\
&+\beta_1\gamma_1 f\left(e_{i-1},e_{j-1}\right)+\beta_2\gamma_1 f\left(v_i,e_{j-1}\right)+\beta_3\gamma_1 f\left(e_i,e_{j-1}\right)+\beta_4\gamma_1 f\left(v_{i+1},e_{j-1}\right)+\beta_0\gamma_2 f\left(v_{i-1},v_j\right)+\beta_1\gamma_2 f\left(e_{i-1},v_j\right)\\
&+\beta_2\gamma_2 f\left(v_i,v_j\right)+\beta_3\gamma_2 f\left(e_i,v_j\right)+\beta_4\gamma_2 f\left(v_{i+1},v_j\right)+\beta_0\gamma_3 f\left(v_{i-1},e_j\right)+\beta_1\gamma_3 f\left(e_{i-1},e_j\right)+\beta_2\gamma_3 f\left(v_i,e_j\right)\\
&+\beta_3\gamma_3 f\left(e_i,e_j\right)+\beta_4\gamma_3 f\left(v_{i+1},e_j\right)+\beta_0\gamma_4 f\left(v_{i-1},v_{j+1}\right)+\beta_1\gamma_4 f\left(e_{i-1},v_{j+1}\right)+\beta_2\gamma_4 f\left(v_i,v_{j+1}\right)\\
&+\beta_3\gamma_4 f\left(e_i,v_{j+1}\right)+\beta_4\gamma_4 f\left(v_{i+1},v_{j+1}\right),\\
\phi_{0,2}=&\alpha_0\beta_0 f\left(v_{i-1},v_j\right)+\alpha_1\beta_0 f\left(v_{i-1},e_j\right)+\alpha_2\beta_0 f\left(v_{i-1},v_{j+1}\right)+\alpha_3\beta_0 f\left(v_{i-1},e_{j+1}\right)+\alpha_4\beta_0 f\left(v_{i-1},v_{j+2}\right)\\
&+\alpha_0\beta_1 f\left(v_{i-1},v_{j+2}\right)+\alpha_1\beta_1 f\left(e_{i-1},e_j\right)\\
&+\alpha_2\beta_1 f\left(e_{i-1},v_{j+1}\right)+\alpha_3\beta_1 f\left(e_{i-1},e_{j+1}\right)+\alpha_4\beta_1 f\left(e_{i-1},v_{j+2}\right)+\alpha_0\beta_2 f\left(v_i,v_j\right)+\alpha_1\beta_2 f\left(v_i,e_j\right)+\alpha_2\beta_2 f\left(v_i,v_{j+1}\right)\\
&+\alpha_3\beta_2 f\left(v_i,e_{j+1}\right)+\alpha_4\beta_2 f\left(v_i,v_{j+2}\right)+\alpha_0\beta_3 f\left(e_i,v_j\right)+\alpha_1\beta_3 f\left(e_i,e_j\right)+\alpha_2\beta_3 f\left(e_i,v_{j+1}\right)+\alpha_3\beta_3 f\left(e_i,e_{j+1}\right)\\
&+\alpha_4\beta_3 f\left(e_i,v_{j+2}\right)+\alpha_0\beta_4 f\left(v_{i+1},v_j\right)+\alpha_1\beta_4 f\left(v_{i+1},e_j\right)+\alpha_2\beta_4 f\left(v_{i+1},v_{j+1}\right)+\alpha_3\beta_4 f\left(v_{i+1},e_{j+1}\right)+\alpha_4\beta_4 f\left(v_{i+1},v_{j+2}\right),\\
\phi_{0,3}=&\beta_0^2 f\left(v_{i-1},v_j\right)+\beta_0\beta_1\left(f\left(e_{i-1},v_j\right)+f\left(v_{i-1},e_j\right)\right)+\beta_1^2 f\left(e_{i-1},e_j\right)+\beta_0\beta_2\left(f\left(v_{i-1},v_{j+1}\right)+f\left(v_i,v_j\right)\right)\\
&+\beta_1\beta_2\left(f\left(e_{i-1},v_{j+1}\right)+f\left(v_i,e_j\right)\right)\\
&+\beta_2^2 f\left(v_i,v_{j+1}\right)+\beta_0\beta_3\left(f\left(e_i,v_j\right)+f\left(v_{i-1},e_{j+1}\right)\right)+\beta_1\beta_3\left(f\left(e_{i-1},e_{j+1}\right)+f\left(e_i,e_j\right)\right)\\
&+\beta_2\beta_3\left(f\left(e_i,v_{j+1}\right)+f\left(v_i,e_{j+1}\right)\right)+\beta_3^2 f\left(e_i,e_{j+1}\right)\\
&+\beta_0\beta_4\left(f\left(v_{i-1},v_{j+2}\right)+f\left(v_{i+1},v_j\right)\right)+\beta_1\beta_4\left(f\left(e_{i-1},v_{j+2}\right)+f\left(v_{i+1},e_j\right)\right)+\beta_2\beta_4\left(f\left(v_i,v_{j+2}\right)+f\left(v_{i+1},v_{j+1}\right)\right)\\
&+\beta_3\beta_4\left(f\left(e_i,v_{j+2}\right)+f\left(v_{i+1},e_{j+1}\right)\right)+\beta_4^2 f\left(v_{i+1},v_{j+2}\right).
\end{aligned}$$

Coefficients $\phi_{i,j}$, $i=1,2,3$ and $j=0,1,2,3$, have similar expressions.

These values of the coefficients correspond to a square having no side on the boundary. For the remainder, a similar procedure using the masks of the univariate quasi-interpolant attached to the boundary leads to similar expressions.

Next, the performance of this nonstandard quasi-interpolation operator is illustrated by considering the Franke and Nielson test functions:

$$\begin{aligned}
g_1(x,y):=&\frac{1}{2}\exp\left(-\left((9x-7)^2+\frac{1}{4}(9y-3)^2\right)\right)+\frac{3}{4}\exp\left(-\frac{1}{49}(9x+1)^2-\frac{1}{10}(9y+1)\right)\\
&-\frac{1}{5}\exp\left(-(9x-4)^2-(9y-7)^2\right)+\frac{3}{4}\exp\left(-\left((9x-2)^2+(9y-2)^2\right)\right),\\
g_2(x,y):=&\frac{1}{2}y\cos^4\left(4\left(x^2+y-1\right)\right).
\end{aligned}$$

Figure 4 shows the plots of Franke and Nielson functions.

Table 2 shows the errors and NCOs for $h=2^{-r}$, $r=4,\ldots,8$. They are in good agreement with the theoretical results.

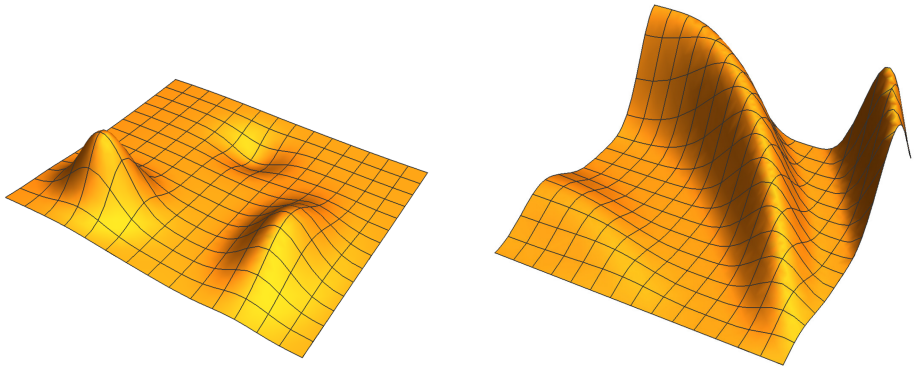

FIGURE 4 From left to right, plots of functions g_1 and g_2 [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 2 Error estimates and NCOs for test functions g_1 and g_2

	g_1		g_2	
n	***error***	***NCO***	***error***	***NCO***
16	0.3625040	–	0.257841	–
32	0.0700742	2.37104	0.0489511	2.39707
64	0.0103237	2.76293	0.00712815	2.77974
128	0.00118445	3.12366	0.000912965	2.96490
256	0.000134193	3.14184	0.000112483	3.02085

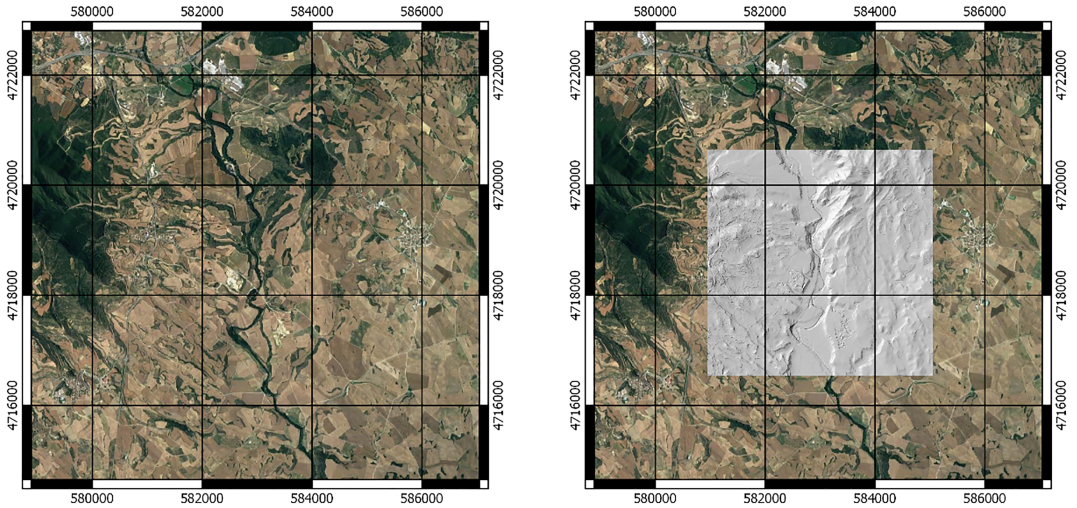

FIGURE 5 Georeferenced area of our DEM (gray area represents the hillshade map covered by the DEM) [Colour figure can be viewed at wileyonlinelibrary.com]

3 | MODELING OF A DEM FROM TENSOR PRODUCT QUASI-INTERPOLATION IN THE BERNSTEIN BASIS

3.1 | Material and methodology

Our DEM reference ($DEM_{ref2\times2}$) will be the *Digital Terrain Model MDT02* produced by the *Instituto Geográfico Nacional* of Spain with a cell size equal to 2 m × 2 m. The coordinate system uses the geodetic reference system ETRS89 and UTM projection UTM zone 30N (see Figure 5). The $DEM_{ref2\times2}$ size is 4096 m × 4096 m centered at the UTM coordinates (583023 m, 4718590 m).

The goodness of the proposed approximation algorithm based on quasi-interpolation in the Bernstein basis is assessed producing a DEM (DEM_{der}) of larger cell sizes than the $DEM_{ref2\times2}$. Then a surface is adjusted and it is generated a resampled DEM (DEM_{res}) with the same cell size than $DEM_{ref2\times2}$ cell. The planimetric and altimetric discrepancies between DEM_{der} and DEM_{res} are calculated separately. The process is divided into the following phases:

1. The $DEM_{ref2\times2}$ is resampled by the closest neighbor method at 3 different resolution levels (4 m × 4 m, 8 m × 8 m and 16 m × 16 m), which produces three different DEM_{resp} which we call, respectively, $DEM_{resp4\times4}$, $DEM_{resp8\times8}$, and $DEM_{resp16\times16}$. They can be represented generically as $DEM_{respX\times X}$.
2. The quasi-interpolation operator fits a surface on each of the $DEM_{respX\times X}$, denoting such a surface as $S_{respX\times X}$.
3. Since the definition domain $S_{respX\times X}$ is the same as $DEM_{ref2\times2}$, $S_{respX\times X}$ can be evaluated at the same points that define the $DEM_{ref2\times2}$ matrix, with which we obtain a homologous $DEM_{der2\times2\_from\_X\times X}$ of the same size as $DEM_{ref2\times2}$.
4. The planimetric quality is assessed using the method based on contour lines introduced in Reinoso[19]: the results given by quasi-interpolation are compared with those produced by the bicubic interpolation.[8]
5. The altimetric quality is assessed by calculating the mean discrepancy in absolute value between the altitudes of $DEM_{ref2\times2}$ and $DEM_{der2\times2\_from\_X\times X}$.

3.2 | Planimetric error assessment

The planimetric error assessment is based on the area enclosed between homologous contours computed on both DEMs, i.e. $DEM_{ref2\times2}$ and $DEM_{der2\times2\_from\_X\times X}$:

1. The contours of both DEMs are computed (Figure 6A,B). Homologous contours are matched automatically (curves $C4_a$ and $C4_b$ in Figure 6A,B). Homologous contours are those with the same height and placed in approximately the same place (see Reinoso[19] to understand how the algorithm automatically eliminate the ambiguity).
2. Overlapping homologous contours (Figure 6C) lead to areas enclosed (gray area in Figure 6D). The planimetric error (Pe_i) referred to the ith pair of homologous contours (C_{i_a}, C_{i_b}) is computed as the surface enclosed between both curves (S_i) divided by the mean length of those contours ($Lm_i = \frac{1}{2}(Li_a + Li_b)$). i.e. $Pe_i = S_i/Lm_i$.
3. The mean planimetric error linked to the $DEM_{der2\times2\_from\_X\times X}$ is denoted as $Pe_{DEMder2x2\_from\_X\times X}$ and its value is the weighted average of the Pe_i of all the homologous contours. The weighting factor is the average length of homologous contours, divided by the total length of the average contours being the total length $L_{Tot} = \sum_{i=1}^{n} Lm_i$, that is,

$$Pe_{DEMder2x2\_from\_X\times X} = \frac{1}{L_{Tot}} \sum_{i=1}^{n} Pe_i \cdot Lm_i.$$

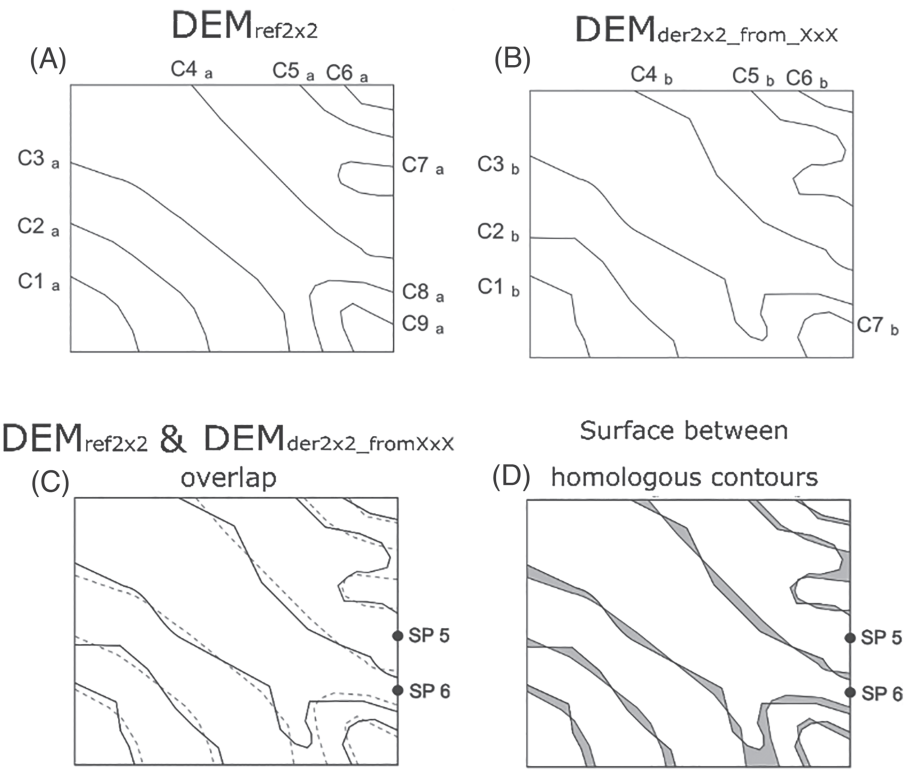

FIGURE 6 Homologous contours and area between them

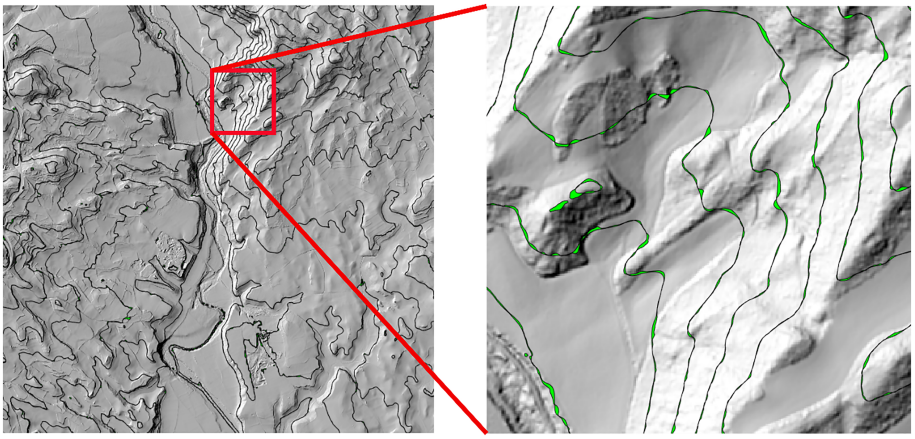

FIGURE 7 Planimetric error contour-based between the $DEM_{ref2\times2}$ and $DEM_{der2\times2\_from\_16\times16}$ [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Planimetric error comparing NS TP QI and the traditional bicubic spline interpolation

$DEM_{der2\times2\_from\_X\times X}$	$Pe_{DEMder2x2\_from\_X\times X}$	NS TP QI (m)	Bicubic (m)
$DEM_{der2\times2\_from\_4\times4}$	Mean	0.08	0.47
	Std	0.01	0.02
$DEM_{der2\times2\_from\_8\times8}$	Mean	0.25	0.55
	Std	0.04	0.03
$DEM_{der2\times2\_from\_16\times16}$	Mean	0.62	0.82
	Std	0.11	0.07

TABLE 4 Altimetric error comparing NS TP QI and the traditional bicubic spline interpolation

$DEM_{der2\times2\_from\_X\times X}$	$Pe_{DEMder2x2\_from\_X\times X}$	NS TP QI (m)	Bicubic (m)
$DEM_{der2\times2\_from\_4\times4}$	Mean	0.03	0.14
	Std	0.05	0.18
$DEM_{der2\times2\_from\_8\times8}$	Mean	0.08	0.16
	Std	0.13	0.22
$DEM_{der2\times2\_from\_16\times16}$	Mean	0.17	0.24
	Std	0.26	0.34

3.3 | Results and discussion

The $DEM_{ref2\times2}$ altitudes range from 376 to 574 m. To study the planimetric error, ten contours have been computed starting at 380 m altitude and ending at 560 m, that is, every 20 m. Figure 7 shows a $DEM_{ref2\times2}$ shadow map overlapping homologous contours and a detailed zone where it can be observed, in green color, the area enclosed by those homologous contours (computed according to the algorithm in Reinoso[19]).

Table 3 shows the planimetric error and the standard deviations for the derived DEMs at resolutions 4×4, 8×8, and 16×16 after applying the proposed non-standard tensor product quasi-interpolation method (abbr. NS TP QI) and comparing it with the traditional bicubic downscaling method.

The approximation method presented here provides lower planimetric errors than bicubic interpolation method in all the resampled resolutions (4×4, 8×8, and 16×16) getting an error reduction of 83%, 54%, and 24%, respectively. On the other hand, the standard deviations are quite similar in both downscaling methods except at the $DEM_{resp16\times16}$ resolution where the standard deviation of our method is greater than that of bicubic interpolation.

The traditional errors study when comparing two homologous DEMs was carried out by comparing the absolute values of errors. If we do not take into account the absolute value, it could give an average value close to zero while the real separation between DEM would be larger, regardless of whether the $DEM_{der2\times2\_from\_X\times X}$ cells are over or under the $DEM_{ref2\times2}$ cells. Table 4 shows the error in absolute value from the different resampled resolutions.

The assessment of altimetric error shows that the proposed algorithm produces better results that the bicubic method for all three DEMs considered. It is observed that the standard deviation is greater than the error and that may be due to the nature of the interpolating bicubic algorithm and the one based on quasi-interpolation.

In order to detect whether our algorithm produces any undesired effect, an error distribution map has been created representing, on a hillshade map, the location of those points where errors are greater than a certain threshold. Figure 8 shows the spatial distribution of altimetric errors greater than 1.25 m from $DEM_{der2\times2\_from\_16\times16}$.

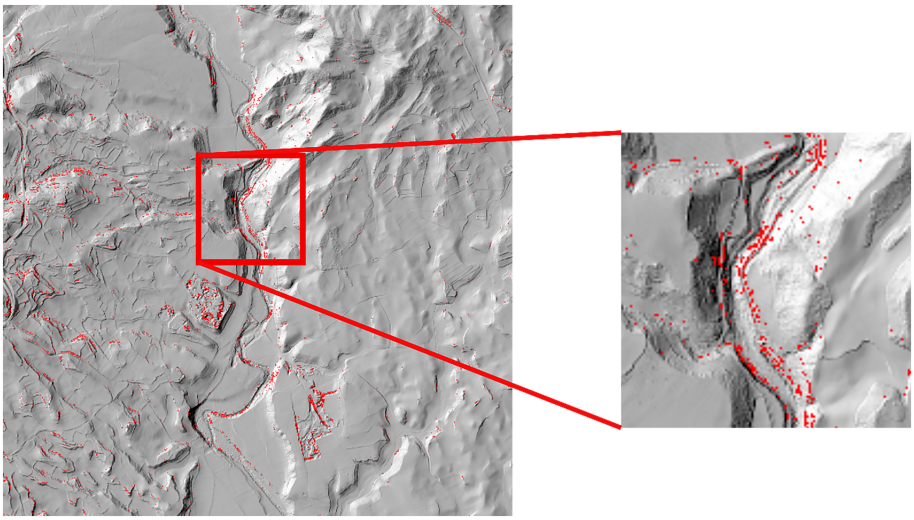

FIGURE 8 Planimetric error contour-based between the DEM2×2 and $DEM_{der2\times2\_from\_16\times16}$ [Colour figure can be viewed at wileyonlinelibrary.com]

Figure 8 shows that the greatest errors accumulate in those places where there are large slopes and a sudden change in slope occurs, such as embankments, rivers, and roads. It is logical that this happens since at the $DEM_{resp16\times16}$ resolution the difference in the cell size with respect to $DEM_{ref2\times2}$ is 14 m and therefore the height difference will also be large. Because the height is represented by one of the extreme values (nearest resample method) it is normal in those areas the error to be larger than in flat areas.

In addition to the best error values presented by our algorithm, it also has the advantage of a lower computational cost with respect to the traditional bicubic one. This advantage places it as a candidate to be implemented in cartographic production software packages.

4 | CONCLUSIONS

In this work, a new quasi-interpolation method has been defined to approximate bivariate surfaces with low computational cost. First, a univariate quasi-interpolant has been defined for functions defined on the real line and modified to approximate functions defined on a bounded interval. The quasi-interpolant is computed on each subinterval by computing its BB-coefficients from some masks independent of the subinterval. Then, it has been used to approximate bivariate functions by a tensor product scheme, and the BB-coefficients of each patch have been determined.

The resulting numerical method has been used to reconstruct the shape of a terrain from a DEM. The C^2-continuity allows to estimate morphological variables that characterize the terrain surface: Slope, orientation, curvature, or normal direction. An immediate application would be the possibility of resampling the approximated surface to obtain DEMs of higher or lower resolution than those used to create the surface.

The approximation method we have proposed has a good performance as the numerical tests have shown. In approximating a DEM, it gets lower planimetric and altimetric errors than the traditional bicubic interpolation algorithm. This can be interpreted as a lower error when downscaling DEM is carried out.

ACKNOWLEDGEMENTS

The authors wish to thank the anonymous referee for his/her very pertinent and useful comments that helped them to improve the original manuscript. The first, second, third, and fifth authors acknowledge the funding of this work by the Spanish State Research Agency (Ministry of Science, Innovation and Universities) under project PID2019-106195RB-I00. The fourth author is a member of the research group FQM 191 Matemática Aplicada funded by the PAIDI programme of the Junta de Andalucía. Funding for open access charge was provided by Universidad de Granada/CBUA.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest regarding the publication of this paper.

ORCID

Salah Eddargani https://orcid.org/0000-0003-4550-1776
María José Ibáñez https://orcid.org/0000-0003-1239-680X

REFERENCES

1. Mesa-Mingorance JL, Ariza-López FJ. Accuracy assessment of Digital Elevation Models (DEMs): a critical review of practices of the past three decades. *Remote Sens*. 2630;2020:12.
2. Shen J, Tan F. Effects of DEM resolution and resampling technique on building treatment for urban inundation modeling: a case study for the 2016 flooding of the HUST campus in Wuhan. *Nat Hazards*. 2020;104:927-957.
3. Leong Tan M, Ficklin DL, Dixon B, Ibrahim AL, Yusop Z, Chaplot V. Impacts of DEM resolution, source, and resampling technique on SWAT-simulated streamflow. *Appl Geogr*. 2015;63:357-368.
4. Gao J. Impact of sampling intervals on the reliability of topographic variables mapped from grid DEMs at a micro-scale. *Int J Geogr Inf Sci*. 1998;12:875-890.
5. Mukherjee S, Joshi PK, Mukherjee S, Ghosh A, Garg RD, Mukhopadhyay A. IU accuracy of open source Digital Elevation Model (DEM). *Int J Appl Earth Obs Geoinf*. 2013;21:205-217.
6. Wang B, Shi W, Liu E. Robust methods for assessing the accuracy of linear interpolated DEM. *Int J Appl Earth Obs Geoinf*. 2015;34:198-206.
7. Maune D. *Digital Elevation Model Technologies and Applications: The DEM User's Manual*. 2nd ed. USA: Bethesda; 2007.
8. Keys R. Cubic convolution interpolation for digital image processing. *IEEE Trans Acoust Speech Signal Process*. 1981;29(6):1153-1160.
9. De Boor C. The quasi-interpolant as a tool in elementary polynomial spline theory. *Approximation Theory I*. New York: Academic Press; 1973:269-276.
10. De Boor C, Fix G. Spline approximation by quasi-interpolants. *J Approx Theory*. 1973;8:19-45.
11. De Boor C. Splines as linear combinations of B-splines, a survey. *Approximation Theory II*. New York: Academic Press; 1976:1-47.
12. Barrera D, Eddargani S, Lamnii A. A novel construction of B-spline-like bases for a family of many knot spline spaces and their application to quasi-interpolation. *J Comput Appl Math*. 2022;404:113761.
13. Barrera D, Eddargani S, Ibáñez MJ, Lamnii A. A new approach to deal with C^2 cubic splines and its application to super-convergent quasi-interpolation. *Math Comput Simul*. 2022;194:401-415.
14. Sorokina T, Zeilfelder F. Optimal quasi-interpolation by quadratic C^1 splines on four-directional meshes. *Approximation Theory, Gatlinburg 2004*. Brentwood: Nashboro Press; 2005:423-438.
15. Sorokina T, Zeilfelder F. An explicit quasi-interpolation scheme based on C^1 quartic splines on type-1 triangulations. *Comput Aided Geom Design*. 2008;25:1-13.
16. Barrera D, Dagnino C, Ibáñez MJ, Remogna S. Point and differential C^1 quasi-interpolation on three direction meshes. *J Comput Appl Math*. 2019;354:373-389.
17. Barrera D, Dagnino C, Ibáñez MJ, Remogna S. Quasi-interpolation by C^1 quartic splines on type-1 triangulations. *J Comput Appl Math*. 2019;349:225-238.
18. Reinoso JF. A priori horizontal displacement (HD) estimation of hydrological features when versioned DEMs are used. *J Hydrol*. 2010;384(1-2):130-141.
19. Reinoso JF. An algorithm for automatically computing the horizontal shift between homologous contours from DTMs. *ISPRS J Photogramm Remote Sens*. 2011;66(3):272-286.

How to cite this article: Ariza-López FJ, Barrera D, Eddargani S, Ibáñez MJ, Reinoso JF. Spline quasi-interpolation in the Bernstein basis and its application to digital elevation models. *Math Meth Appl Sci*. 2023;46(2):1687-1698. doi:10.1002/mma.8602