Research Article

Progressive Iterative Approximation for Extended B-Spline Interpolation Surfaces

Yeqing Yi ¹, Zixuan Tang ², and Chengzhi Liu ²

¹School of Information, Hunan University of Humanities, Science and Technology, Loudi 417000, China

²School of Mathematics and Finance, Hunan University of Humanities, Science and Technology, Loudi 417000, China

Correspondence should be addressed to Chengzhi Liu; it-rocket@163.com

Received 8 February 2021; Revised 23 March 2021; Accepted 13 April 2021; Published 27 April 2021

Academic Editor: Giovanni Garcea

Copyright © 2021 Yeqing Yi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to improve the computational efficiency of data interpolation, we study the progressive iterative approximation (PIA) for tensor product extended cubic uniform B-spline surfaces. By solving the optimal shape parameters, we can minimize the spectral radius of PIA's iteration matrix, and hence the convergence rate of PIA is accelerated. Stated numerical examples show that the optimal shape parameters make the PIA have the fastest convergence rate.

1. Introduction

Data interpolation plays important roles in scientific research and engineering applications. How to solve interpolation curves/surfaces efficiently has been one of the most popular topics in computer-aided geometric design (see [1–3]). Oftentimes, one has to solve a linear system to obtain the interpolation curves or surfaces. Efficient and accurate algorithms are required to guarantee the computational efficiency. For small-scale systems, direct methods are typically the preferred choices. However, for large-scale systems, it becomes necessary to employ iterative methods to obtain the solutions. In recent years, an iterative method, namely, progressive iterative approximation (PIA), has attracted a lot of attention and has become a very hot research area. The PIA stands out because it has the advantages of clear geometric meaning, stable convergence, simple iterative format, local modification, and so on. Furthermore, it avoids to solve a linear system directly. For more details about PIA, we refer the readers to read a recent survey [4].

Despite the fact that the PIA offers many advantages, there is a disadvantage, that is, slow rate of convergence. To overcome this limitation and further improve the computational efficiency, a great deal of acceleration techniques have been conducted. Examples of such approaches include [5–14] and a lot of literatures therein.

The emergence of blending bases with shape parameters has enriched the theories and methods of geometric modeling [1, 15–17]. Due to the flexibility in shape adjustment, splines with shape parameters have drawn much attention for decades and a large number of splines with shape parameters were exploited (see, for example, [18–20]). Very often, the aim of shape parameters is to adjust the shapes of splines, while in [21], the introduction of shape parameters is to speed up the convergence rate of PIA. In that paper, the eigenvalues of the collocation matrix were expressed explicitly, and hence the optimal shape parameters were solved to make the PIA have the fastest convergence rate. Based on this conclusion, we further study the PIA format for tensor product extended cubic uniform B-spline surfaces, which is an extension of the PIA for the classic bicubic uniform B-spline curves. By solving the optimal shape parameters, the convergence rate of PIA is accelerated, and thus the computational efficiency of data interpolation can be improved.

The rest of this paper is organized as follows. After recapping the definition of the extended cubic uniform B-spline surfaces with shape parameters, we exploit the PIA format for extended cubic uniform B-spline surfaces in Section 2. In Section 3, we study the optimal shape parameters to make the PIA have the fastest convergence rate. Some numerical examples are given to illustrate the

acceleration effect in Section 4. Finally, we give some concluding remarks in Section 5.

2. PIA for Extended Cubic Uniform B-Spline Surface

2.1. Extended Cubic Uniform B-Spline Surface. We begin with the definition of the extended cubic uniform B-spline basis with a shape parameter.

$$\left\{ \begin{array}{l} N_0(t; \lambda) = \frac{1}{6+2\lambda} (1-\lambda t)(1-t)^3, \\ N_1(t; \lambda) = \frac{2+\lambda}{3+\lambda} (1-\lambda t)(1-t)^3 + \frac{2+\lambda}{3+\lambda} (3+\lambda-\lambda t)(1-t)^2 t + \frac{1}{3+\lambda} (3+\lambda t)(1-t)t^2 + \frac{1}{6+2\lambda} (1-\lambda+\lambda t)t^3, \\ N_2(t; \lambda) = \frac{1}{6+2\lambda} (1-\lambda t)(1-t)^3 + \frac{1}{3+\lambda} (3+\lambda-\lambda t)(1-t)^2 t + \frac{2+\lambda}{3+\lambda} (3+\lambda t)(1-t)t^2 + \frac{2+\lambda}{3+\lambda} (1-\lambda+\lambda t)t^3, \\ N_3(t; \lambda) = \frac{1}{6+2\lambda} (1-\lambda+\lambda t)t^3, \end{array} \right. \quad (1)$$

where $\lambda \in (-1, 1]$ is the so-called shape parameter.

The λ -B-spline basis has the properties of non-negativity and symmetry, and it will degenerate into the classic cubic B-spline basis if $\lambda = 0$ [22].

Definition 2 (see [22]). Given knot vectors $(u_1, u_2, \dots, u_{m-1})$ and $(v_1, v_2, \dots, v_{n-1})$ such that $u_1 < u_2 < \dots < u_{m-1}$

Definition 1 (see [22]). For $0 \leq t \leq 1$, the extended cubic uniform B-spline basis (abbr. λ -B-spline basis) is

and $v_1 < v_2 < \dots < v_{n-1}$. Let $\mathbf{p}_{ij} \in \mathbb{R}^3$ ($i = 0, 1, \dots, m$; $j = 0, 1, \dots, n$) be the control points, and Let $\{N_l(t; \lambda_1)\}_{l=0}^3$ and $\{N_s(t; \lambda_2)\}_{s=0}^3$ be the λ -B-spline bases defined as in (1). Then, for $u \in [u_i, u_{i+1}]$, $i = 1, 2, \dots, m-2$ and $v \in [v_j, v_{j+1}]$, $j = 1, 2, \dots, n-2$, we can define $(m-2) \times (n-2)$ extended cubic uniform B-spline patches with shape parameters λ_1 and λ_2 as

$$\mathbf{S}_{ij}(u, v) = \sum_{l=0}^3 \sum_{s=0}^3 \mathbf{p}_{i+l-1, j+s-1} N_l\left(\frac{u-u_i}{u_{i+1}-u_i}; \lambda_1\right) N_s\left(\frac{v-v_j}{v_{j+1}-v_j}; \lambda_2\right). \quad (2)$$

All these patches comprise an entire extended cubic uniform B-spline surface (abbr. λ -B-spline surface):

$$\mathbf{S}(u, v) = \mathbf{S}_{ij}(u, v), \quad i = 1, 2, \dots, m-2; j = 1, 2, \dots, n-2. \quad (3)$$

Due to the degeneracy property of the λ -B-spline basis, it is easy to verify that the λ -B-spline surface will degenerate into the classic bicubic B-spline surface if $\lambda_1 = \lambda_2 = 0$.

If we want the λ -B-spline surface to interpolate the boundary control points, we have to add several control vertices according to

$$\left\{ \begin{array}{ll} \mathbf{p}_{-1,-1} = 2\mathbf{p}_{0,0} - \mathbf{p}_{1,1}, & \mathbf{p}_{-1,n+1} = 2\mathbf{p}_{0,n} - \mathbf{p}_{1,n-1}, \\ \mathbf{p}_{m+1,-1} = 2\mathbf{p}_{m,0} - \mathbf{p}_{m-1,1}, & \mathbf{p}_{m+1,n+1} = 2\mathbf{p}_{m,n} - \mathbf{p}_{m-1,n-1}, \\ \mathbf{p}_{i,-1} = 2\mathbf{p}_{i,0} - \mathbf{p}_{i,1}, & \mathbf{p}_{i,n+1} = 2\mathbf{p}_{i,n} - \mathbf{p}_{i,n-1}, \quad i = 0, 1, \dots, n, \\ \mathbf{p}_{-1,j} = 2\mathbf{p}_{0,j} - \mathbf{p}_{1,j}, & \mathbf{p}_{m+1,j} = 2\mathbf{p}_{m,j} - \mathbf{p}_{m-1,j}, \quad j = 0, 1, \dots, m. \end{array} \right. \quad (4)$$

2.2. *PIA*. Let $\{N_l(t; \lambda_1)\}_{l=0}^3$ and $\{N_s(t; \lambda_2)\}_{s=0}^3$ be the λ -B-spline bases defined as in (1). Given a set of organized data points $\{\mathbf{p}_{ij}\}_{i=0, \dots, m}^{j=0, \dots, n}$ to be interpolated, we assign a pair of parameters (u_i, v_j) to the (i, j) th point $\mathbf{p}_{ij}$, where $u_0 < u_1 < \dots < u_m$ and $v_0 < v_1 < \dots < v_n$.

Firstly, the points $\{\mathbf{p}_{ij}\}_{i=0, \dots, m}^{j=0, \dots, n}$ as well as these added points (4) are interpreted as the control points of a λ -B-spline surface. Therefore, we can obtain the initial approximate interpolation curve

$$\mathbf{S}^{(0)}(u, v) = \mathbf{S}_{ij}^{(0)}(u, v) = \sum_{l=0}^3 \sum_{s=0}^3 \mathbf{p}_{i+l-1, j+s-1}^{(0)} N_l\left(\frac{u-u_i}{u_{i+1}-u_i}; \lambda_1\right) N_s\left(\frac{v-v_j}{v_{j+1}-v_j}; \lambda_2\right), \quad i = 0, 1, \dots, m-1; j = 0, 1, \dots, n-1, \quad (5)$$

where $\mathbf{p}_{ij}^{(0)} = \mathbf{p}_{ij}$, $i = -1, 0, \dots, m+1$; $j = -1, 0, \dots, n+1$. Secondly, let

$$\delta_{ij}^{(0)} = \mathbf{p}_{ij} - \mathbf{S}^{(0)}(u_i, v_j), \quad i = -1, 0, \dots, m+1; j = -1, 0, \dots, n+1 \quad (6)$$

be the adjusting vectors of the control points. Then, we can adjust the control points according to

$$\mathbf{p}_{ij}^{(1)} = \mathbf{p}_{ij}^{(0)} + \delta_{ij}^{(0)}, \quad i = -1, 0, \dots, m+1; j = -1, 0, \dots, n+1. \quad (7)$$

Suppose that we have obtained the $(k-1)$ th, $k = 1, 2, \dots$, approximate interpolation surface $\mathbf{S}^{(k-1)}(u, v)$; then, the k th approximate interpolation surface can be defined as

$$\mathbf{S}^{(k)}(u, v) = \mathbf{S}_{ij}^{(k)}(u, v) = \sum_{l=0}^3 \sum_{s=0}^3 \mathbf{p}_{i+l-1, j+s-1}^{(k)} N_l\left(\frac{u-u_i}{u_{i+1}-u_i}; \lambda_1\right) N_s\left(\frac{v-v_j}{v_{j+1}-v_j}; \lambda_2\right), \quad i = 0, 1, \dots, m-1; j = 0, 1, \dots, n-1, \quad (8)$$

where

$$\mathbf{p}_{ij}^{(k)} = \mathbf{p}_{ij}^{(k-1)} + \delta_{ij}^{(k-1)}, \quad \delta_{ij}^{(k-1)} = \mathbf{p}_{ij} - \mathbf{S}^{(k-1)}(u_i, v_j), \quad i = -1, 0, \dots, m+1; j = -1, 0, \dots, n+1. \quad (9)$$

Therefore, we obtain a sequence of approximate interpolation surfaces $\mathbf{S}^{(k)}(u, v)$, $k = 0, 1, \dots$. The initial surface $\mathbf{S}^{(0)}(u, v)$ is said to have the PIA property if the limit of $\mathbf{S}^{(k)}(u, v)$ interpolates the points $\{\mathbf{p}_{ij}\}_{i=0, \dots, m}^{j=0, \dots, n}$. It was shown in [23] that tensor product surfaces generated by normalized

and totally positive bases have the PIA property. We note in [22] that the λ -B-spline basis is normalized and totally positive; therefore, the initial λ -B-spline surface has the PIA property.

Let

$$\mathbf{P} = [\mathbf{p}_{-1, -1}, \mathbf{p}_{-1, 0}, \dots, \mathbf{p}_{-1, n+1}, \mathbf{p}_{0, -1}, \dots, \mathbf{p}_{0, n+1}, \dots, \mathbf{p}_{m+1, -1}, \dots, \mathbf{p}_{m+1, n+1}]^T, \quad (10)$$

$$\mathbf{P}^{(k)} = [\mathbf{p}_{-1, -1}^{(k)}, \mathbf{p}_{-1, 0}^{(k)}, \dots, \mathbf{p}_{-1, n+1}^{(k)}, \mathbf{p}_{0, -1}^{(k)}, \dots, \mathbf{p}_{0, n+1}^{(k)}, \dots, \mathbf{p}_{m+1, -1}^{(k)}, \dots, \mathbf{p}_{m+1, n+1}^{(k)}]^T. \quad (11)$$

Then, equation (9) can be written as

$$\mathbf{P}^{(k+1)} = (I - B_1 \otimes B_2) \mathbf{P}^{(k)} + \mathbf{P}, \quad (12)$$

where $\otimes$ is the Kronecker product, I is the identity matrix, and $B_1 \in \mathbb{R}^{(m+3) \times (m+3)}$ and $B_2 \in \mathbb{R}^{(n+3) \times (n+3)}$ are the collocation matrices resulting from the λ -B-spline basis; in detail,

$$\begin{aligned}
 B_1 &= \begin{bmatrix} 1 & 0 & & & \\ \frac{1}{6+2\lambda_1} & \frac{2+\lambda_1}{3+\lambda_1} & \frac{1}{6+2\lambda_1} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{1}{6+2\lambda_1} & \frac{2+\lambda_1}{3+\lambda_1} & \frac{1}{6+2\lambda_1} \\ & & & 0 & 1 \end{bmatrix}, \\
 B_2 &= \begin{bmatrix} 1 & 0 & & & \\ \frac{1}{6+2\lambda_2} & \frac{2+\lambda_2}{3+\lambda_2} & \frac{1}{6+2\lambda_2} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{1}{6+2\lambda_2} & \frac{2+\lambda_2}{3+\lambda_2} & \frac{1}{6+2\lambda_2} \\ & & & 0 & 1 \end{bmatrix}.
 \end{aligned} \tag{13}$$

We refer to the iterative format (12) as the PIA format. The matrix $I - B_1 \otimes B_2$ in (12) is the iteration matrix. It is well known that the PIA converges if and only if the spectral radius of the iteration matrix $I - B_1 \otimes B_2$ is less than 1. Moreover, the smaller the spectral radius is, the faster the PIA converges. According to (13), the spectral radius is a function of the shape parameters λ_1 and λ_2 . This indicates that we can minimize the spectral radius by selecting optimal shape parameters λ_1 and λ_2 , and hence the convergence of PIA can be improved. We will discuss the optimal shape parameters λ_1 and λ_2 in the following section.

Finally, we give some notations. The spectral radius of a matrix A is denoted by $\rho(A)$, the eigenvalues of A are denoted by $\mu_i(A)$, and the smallest eigenvalue of a matrix A is denoted by $\mu_{\min}(A)$.

3. Optimal Shape Parameters

In order to make the PIA have the fastest convergence rate, we have to solve the optimal shape parameters λ_1 and λ_2 that minimize the spectral radius of PIA's iteration matrix, i.e.,

$$\min_{\lambda_1, \lambda_2 \in (-1, 1]} \rho(I - B_1 \otimes B_2). \tag{14}$$

Lemma 1 (see [24]). Suppose that A and B are square matrices of size m and n , respectively. Let $\{\mu_i(A)\}_{i=1}^m$ and $\{\mu_j(B)\}_{j=1}^n$ be the eigenvalues of A and B , respectively. Then, the eigenvalues of $A \otimes B$ are $\mu_i(A)\mu_j(B)$, $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$.

Lemma 2 (see [21]). Let B be an $n \times n$ collocation matrix resulting from the λ -B-spline basis. Then, the eigenvalues of B are

$$\mu_i(B) = \begin{cases} \frac{2+\lambda}{3+\lambda} + \frac{1}{3+\lambda} \cos\left(\frac{i}{n-1}\pi\right), & i = 0, 1, \dots, n-2. \\ 1, & i = n-1. \end{cases} \tag{15}$$

By direct deduction, we have the following corollary.

Corollary 1. Let B be an $n \times n$ collocation matrix resulting from the λ -B-spline basis. Then,

- (1) The eigenvalues of B are distributed at the interval $(0, 1]$.
- (2) Given $\lambda \in (0, 1]$, the smallest eigenvalue of B is $\mu_{\min}(B) = 2 + \lambda/3 + \lambda + 1/3 + \lambda \cos(n-2/n-1\pi)$.
- (3) $\mu_{\min}(B)$ decreases as λ increases and reaches minimum when $\lambda = 1$.

Theorem 1. Let $B_1 \in \mathbb{R}^{(m+3) \times (m+3)}$ and $B_2 \in \mathbb{R}^{(n+3) \times (n+3)}$ be the collocation matrices defined as in (13). For fixed $\lambda_1, \lambda_2 \in (-1, 1]$, the spectral radius of the iteration matrix of PIA is

$$\rho(I - B_1 \otimes B_2) = 1 - \frac{[2 + \lambda_1 + \cos(m + 1/m + 2\pi)][2 + \lambda_2 + \cos(n + 1/n + 2\pi)]}{(3 + \lambda_1)(3 + \lambda_2)}. \tag{16}$$

The PIA has the fastest convergence rate when $\lambda_1 = \lambda_2 = 1$, and in such case, the spectral radius is

$$\rho(I - B_1 \otimes B_2) = 1 - \frac{[3 + \cos(m + 1/m + 2\pi)][3 + \cos(n + 1/n + 2\pi)]}{16}. \tag{17}$$

Proof. According to Corollary 1, for $i = 0, 1, \dots, m+2$; $j = 0, 1, \dots, n+2$, we have $0 < \mu_i(B_1), \mu_j(B_2) \leq 1$, so is the

product of $\mu_i(B_1)$ and $\mu_j(B_2)$, i.e., $0 < \mu_i(B_1)\mu_j(B_2) \leq 1$. Combined with Lemma 1, we have

$$\begin{aligned} \rho(I - B_1 \otimes B_2) &= 1 - \mu_{\min}(B_1 \otimes B_2) = 1 - \mu_{\min}(B_1)\mu_{\min}(B_2) \\ &= 1 - \frac{[2 + \lambda_1 + \cos(m + 1/m + 2\pi)][2 + \lambda_2 + \cos(n + 1/n + 2\pi)]}{(3 + \lambda_1)(3 + \lambda_2)}. \end{aligned} \quad (18)$$

From Corollary 1, $\mu_{\min}(B_1)$ and $\mu_{\min}(B_2)$ minimize at $\lambda_1 = 1$ and $\lambda_2 = 1$, respectively. By substituting $\lambda_1 = \lambda_2 = 1$ into (16), the result (17) follows straightforwardly. $\square$

4. Numerical Examples

In this section, several numerical examples are presented to assess the effectiveness of the optimal shape parameters. All experiments were performed by Matlab R2012b.

Let $\{\mathbf{p}_{ij}\}_{i=0, \dots, m}^{j=0, \dots, n}$ be the points to be interpolated, and let $\mathbf{S}^{(k)}(u, v)$ be the k th approximate interpolation λ -B-spline

surface. Then, the interpolation error of $\mathbf{S}^{(k)}(u, v)$ can be defined as

$$\varepsilon^{(k)} = \max_{0 \leq i \leq m, 0 \leq j \leq n} \|\delta_{ij}^{(k)}\| = \max_{0 \leq i \leq m, 0 \leq j \leq n} \|\mathbf{p}_{ij} - \mathbf{S}^{(k)}(u_i, v_j)\|, \quad (19)$$

where $\|\cdot\|$ is the Euclidean norm.

Example 1. Consider the data interpolation of 11×11 points $\{\mathbf{p}_{ij}\}_{i=0,1, \dots, 10}^{j=0,1, \dots, 10}$ sampled from the peaks function

$$f(x, y) = 3(1-x)^2 e^{-x^2-(y+1)^2} - 10\left(\frac{1}{5}x - x^3 - y^5\right) e^{-x^2-y^2} - \frac{1}{3}e^{-(x+1)^2-y^2}, \quad (x, y) \in [-3, 3] \otimes [-4, 4], \quad (20)$$

in the following way:

$$\mathbf{p}_{ij} = \left(-3 + \frac{3}{5}i, -4 + \frac{4}{5}j, f\left(-3 + \frac{3}{5}i, -4 + \frac{4}{5}j\right)\right), \quad i, j = 0, 1, \dots, 10. \quad (21)$$

Example 2. Consider the data interpolation of 16 points: $(1, 1, 1), (2, 1, 2), (3, 1, 2), (4, 1, 1), (1, 2, 2), (2, 2, 2.5), (3, 2, 2.5), (4, 2, 2), (2, 3, 2), (1, 3, 2), (2, 3, 2.5), (4, 3, 2), (1, 4, 1), (2, 4, 2), (3, 4, 2), (4, 4, 1)$.

Example 3. Consider the data interpolation of 90×100 points $\{\mathbf{p}_{ij}\}_{i=0,1, \dots, 89}^{j=0,1, \dots, 99}$ sampled from the function

$$f(x, y) = \frac{3}{4}e^{-(9x-2)^2+(9y-2)^2/4} + \frac{3}{4}e^{-(9x+1)^2/49-9y+1/10} + \frac{1}{2}e^{-(9x-7)^2+(9y-3)^2/4} - \frac{1}{5}e^{-(9x-4)^2-(9y-7)^2}, \quad (22)$$

at $\{(x_i, y_j)\}: x_i = i/89, i = 0, 1, \dots, 89; y_j = j/99, j = 0, 1, \dots, 99$.

Example 4. Consider the data interpolation of 160×160 points $\{\mathbf{p}_{ij}\}_{i=0,1, \dots, 159}^{j=0,1, \dots, 159}$ sampled from the function

$$f(x, y) = \sin\left(\sqrt{x^2 + y^2}\right)/\sqrt{x^2 + y^2}, \quad (23)$$

at $\{(x_i, y_j)\}: x_i = -8 + 16i/159, y_j = -8 + 16j/159, i, j = 0, 1, \dots, 159$.

The PIA for λ -B-spline surfaces with different λ_1 and λ_2 is employed to interpolate the points in Examples 1–4. It should be pointed out that the PIA for λ -B-spline surfaces will degenerate into the PIA for the classic bicubic B-spline surfaces if $\lambda_1 = \lambda_2 = 0$.

As an illustration, we show in Figure 1 the spectral radii of PIA's iteration matrices with different shape parameters

$\lambda_1, \lambda_2 \in (0, 1]$ in Examples 1–4. In Table 1, we list the spectral radii of PIA's iteration matrices. For convenience, the notation (λ_1, λ_2) in Table 1 and the subsequent tables represents the values of the shape parameters λ_1 and λ_2 . We can see from Figure 1 and Table 1 that the spectral radii of iteration matrices are less than 1 for any $\lambda_1, \lambda_2 \in (-1, 1]$ and minimize at $\lambda_1 = \lambda_2 = 1$, and hence the PIA converges for $\lambda_1, \lambda_2 \in (-1, 1]$ and has the fastest convergence rate when $\lambda_1 = \lambda_2 = 1$. Those results coincide with the conclusions in Theorem 1. Thus, for the optimal shape parameters λ_1 and λ_2 , the convergence rate of PIA for λ -B-spline surfaces would achieve a great acceleration compared with that for the classic bicubic B-spline surfaces.

Given interpolation errors, we list in Table 2 the number of required iterations when we test Example 1 with different shape parameters. It is evident from Table 2 that under the requirement of the same precision, the number of iterations

FIGURE 1: Spectral radii of PIA's iteration matrices with different λ_1 and λ_2 in Examples 1–4. (a) Example 1. (b) Example 2. (c) Example 3. (d) Example 4.

TABLE 1: Spectral radii of PIA's iteration matrices with different (λ_1, λ_2) in Examples 1–4.

Example	$\rho(I - B_1 \otimes B_2)$								
	$(-1/2, -1/2)$	$(-1/2, 0)$	$(-1/2, 1/2)$	$(-1/2, 1)$	$(0, 0)$	$(0, 1)$	$(1, 0)$	$(1/2, 1)$	$(1, 1)$
Example 1	0.8400	0.8000	0.7714	0.7500	0.7500	0.6875	0.6875	0.6429	0.6094
Example 2	0.9518	0.9232	0.9028	0.8875	0.8777	0.8209	0.8209	0.7733	0.7376
Example 3	0.9599	0.9332	0.9142	0.8999	0.8888	0.8332	0.8332	0.7856	0.7499
Example 4	0.9600	0.9333	0.9142	0.9000	0.8888	0.8333	0.8333	0.7857	0.7500

TABLE 2: Number of required PIA iterations for fixed interpolation error in Example 1.

ε	$(-1/2, -1/2)$	$(-1/2, 0)$	$(-1/2, 1/2)$	$(-1/2, 1)$	$(0, 0)$	$(0, 1)$	$(1, 0)$	$(1/2, 1)$	$(1, 1)$
$1e-03$	43	30	26	24	21	16	17	12	12
$1e-04$	76	49	40	36	33	24	24	19	16
$1e-05$	114	72	58	50	47	32	33	26	22
$1e-06$	152	96	77	66	63	42	43	33	28
$1e-07$	191	122	97	84	78	52	54	41	35
$1e-08$	232	147	118	102	94	63	65	49	42
$1e-09$	273	174	139	120	110	73	76	57	49
$1e-10$	317	200	160	139	126	84	86	66	56

of PIA with $\lambda_1 = \lambda_2 = 1$ is the smallest. In Tables 3 and 4, we list the interpolation errors of Examples 2–4 obtained by implementing the PIA for λ -B-spline surfaces with different λ_1 and λ_2 . We can see that with the same iterations, the

interpolation errors obtained by the PIA with $\lambda_1 = \lambda_2 = 1$ are the smallest. Figures 2–9 display the λ -B-spline surfaces with different shape parameters when we employ the PIA to interpolate the data given in Examples 1–4.

TABLE 3: Interpolation errors of PIA in Example 2 with different (λ_1, λ_2) .

k	$(-1/2, -1/2)$	$(-1/2, 0)$	$(-1/2, 1/2)$	$(-1/2, 1)$	$(0, 0)$	$(0, 1)$	$(1, 0)$	$(1/2, 1)$
0	$2.20e-01$	$2.00e-01$	$1.86e-01$	$1.75e-01$	$1.81e-01$	$1.56e-01$	$1.56e-01$	$1.33e-01$
1	$1.52e-02$	$1.22e-02$	$1.10e-02$	$1.06e-02$	$8.87e-03$	$6.73e-03$	$6.73e-03$	$3.78e-03$
2	$7.33e-03$	$5.89e-03$	$4.61e-03$	$3.55e-03$	$5.00e-03$	$3.35e-03$	$3.35e-03$	$2.53e-03$
3	$5.20e-03$	$3.80e-03$	$2.86e-03$	$2.21e-03$	$2.82e-03$	$1.67e-03$	$1.67e-03$	$1.02e-03$
4	$2.38e-03$	$1.61e-03$	$1.15e-03$	$8.64e-04$	$1.07e-03$	$5.69e-04$	$5.69e-04$	$2.93e-04$
5	$9.60e-04$	$6.00e-04$	$4.09e-04$	$2.97e-04$	$3.64e-04$	$1.72e-04$	$1.72e-04$	$7.52e-05$
10	$6.54e-06$	$2.80e-06$	$1.46e-06$	$8.65e-07$	$1.08e-06$	$2.85e-07$	$2.85e-07$	$5.84e-08$
15	$3.98e-08$	$1.16e-08$	$4.52e-09$	$2.15e-09$	$2.89e-09$	$4.19e-10$	$4.19e-10$	$4.14e-11$
20	$2.41e-10$	$4.78e-11$	$1.39e-11$	$5.23e-12$	$7.69e-12$	$6.10e-13$	$6.10e-13$	$2.89e-14$

TABLE 4: Interpolation errors of PIA in Examples 3 and 4 with different (λ_1, λ_2) .

k	Example 3			Example 4		
	$(-1/2, -1/2)$	$(0, 0)$	$(1, 1)$	$(-1/2, -1/2)$	$(0, 0)$	$(1, 1)$
0	$1.48e-03$	$1.24e-03$	$9.27e-04$	$1.35e-03$	$1.12e-03$	$8.42e-04$
1	$4.87e-05$	$3.39e-05$	$1.91e-05$	$4.71e-05$	$3.27e-05$	$1.84e-05$
2	$1.94e-05$	$1.12e-05$	$4.75e-06$	$1.88e-05$	$1.09e-05$	$4.58e-06$
3	$9.67e-06$	$4.67e-06$	$1.48e-06$	$9.37e-06$	$4.52e-06$	$1.43e-06$
4	$5.41e-06$	$2.18e-06$	$5.17e-07$	$5.24e-06$	$2.11e-06$	$5.00e-07$
5	$3.70e-06$	$1.24e-06$	$2.21e-07$	$3.59e-06$	$1.20e-06$	$2.14e-07$
10	$6.22e-07$	$8.39e-08$	$3.55e-09$	$6.04e-07$	$8.13e-08$	$3.43e-09$
20	$3.37e-08$	$7.35e-10$	$1.76e-12$	$3.27e-08$	$7.11e-10$	$1.69e-12$
50	$2.09e-11$	$1.21e-13$	$3.15e-16$	$1.67e-11$	$7.89e-14$	$1.99e-15$

FIGURE 2: Interpolation surfaces obtained by PIA with $\lambda_1 = \lambda_2 = 0$ at the k th iteration in Example 1. (a) $k=0$. (b) $k=1$. (c) $k=10$. FIGURE 3: Interpolation surfaces obtained by PIA with $\lambda_1 = \lambda_2 = 1$ at the k th iteration in Example 1. (a) $k=0$. (b) $k=1$. (c) $k=10$.

FIGURE 4: Interpolation surfaces obtain by PIA with $\lambda_1 = \lambda_2 = 0$ at the k th iteration in Example 2. (a) $k=0$. (b) $k=1$. (c) $k=5$.

FIGURE 5: Interpolation surfaces obtain by PIA with $\lambda_1 = \lambda_2 = 1$ at the k th iteration in Example 2. (a) $k=0$. (b) $k=1$. (c) $k=5$.

FIGURE 6: Interpolation surfaces obtained by PIA with $\lambda_1 = \lambda_2 = 0$ at the k th iteration in Example 3. (a) $k=0$. (b) $k=2$.

FIGURE 7: Interpolation surfaces obtained by PIA with $\lambda_1 = \lambda_2 = 1$ at the k th iteration in Example 3. (a) $k=0$. (b) $k=2$.

FIGURE 8: Interpolation surfaces obtained by PIA with $\lambda_1 = \lambda_2 = 0$ at the k th iteration in Example 4. (a) $k=0$. (b) $k=2$.

FIGURE 9: Interpolation surfaces obtained by PIA with $\lambda_1 = \lambda_2 = 1$ at the k th iteration in Example 4. (a) $k=0$. (b) $k=2$.

All these numerical results shown in these tables and figures indicate that the PIA for λ -B-spline surfaces has the fastest convergence rate when $\lambda_1 = \lambda_2 = 1$.

5. Conclusion

In this paper, we have exploited the PIA format for λ -B-spline surfaces. Due to the introduction of shape parameters, we can make the PIA have the fastest convergence rate by solving the optimal shape parameters, while the amount of calculation does not increase. Therefore, it inherits the merits of the PIA for the classic bicubic B-spline surfaces, e.g., simple iterative scheme, stable convergence, clear geometric meaning, local modification, etc. More importantly, the computational efficiency of data interpolation is improved by accelerating the convergence rate.

Data Availability

The data are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the Natural Science Foundation of Hunan Province (grant no. 2020JJ5267) and Scientific Research Funds of Hunan Provincial Education Department (grant nos. CX20201192 and 19B301).

References

- [1] Y. Zhu and X. Han, "C² rational quartic interpolation spline with local shape preserving property," *Applied Mathematics Letters*, vol. 46, pp. 57–63, 2015.
- [2] X. Han, "Shape-preserving piecewise rational interpolation with higher order continuity," *Applied Mathematics and Computation*, vol. 337, pp. 1–13, 2018.
- [3] M. Abbas, A. A. Majid, and J. M. Ali, "Positivity-preserving rational bi-cubic spline interpolation for 3D positive data," *Applied Mathematics and Computation*, vol. 234, pp. 460–476, 2014.
- [4] H. Lin, T. Maekawa, and C. Deng, "Survey on geometric iterative methods and their applications," *Computer-Aided Design*, vol. 95, pp. 40–51, 2017.
- [5] L. Lu, "Weighted progressive iteration approximation and convergence analysis," *Computer Aided Geometric Design*, vol. 27, no. 2, pp. 129–137, 2010.
- [6] J. M. Carnicer, J. Delgado, and J. Peña, "Richardson method and totally nonnegative linear systems," *Linear Algebra and Its Applications*, vol. 11, 2010.
- [7] J. M. Carnicer, J. Delgado, and J. M. Peña, "On the progressive iteration approximation property and alternative iterations," *Computer Aided Geometric Design*, vol. 28, no. 9, pp. 523–526, 2011.
- [8] S. Deng and G. Wang, "Numerical analysis of the progressive iterative approximation method," *Computer Aided Geometric Design*, vol. 7, pp. 879–884, 2012.
- [9] C. Liu, X. Han, and J. Li, "The Chebyshev accelerating method for progressive iterative approximation," *Communications in Information and Systems*, vol. 17, no. 1, pp. 25–43, 2017.
- [10] L. Zhang, J. Tan, X. Ge, and G. Zheng, "Generalized B-splines' geometric fitting method with mutually different weights," *Journal of Computational and Applied Mathematics*, vol. 329, pp. 331–343, 2018.
- [11] A. Ebrahimi and G. B. Loghmani, "A composite iterative procedure with fast convergence rate for the progressive-iteration approximation of curves," *Journal of Computational and Applied Mathematics*, vol. 359, pp. 1–15, 2019.
- [12] C. Liu and Z. Liu, "Progressive iterative approximation with preconditioners," *Mathematics*, vol. 8, no. 9, p. 1503, 2020.
- [13] C. Liu, X. Han, and J. Li, "Preconditioned progressive iterative approximation for triangular Bézier patches and its application," *Journal of Computational and Applied Mathematics*, vol. 366, no. 112389, 2020.
- [14] C. Liu, Z. Liu, and X. Han, "Preconditioned progressive iterative approximation for tensor product Bézier patches," *Mathematics and Computers in Simulation*, vol. 185, pp. 372–383, 2021.
- [15] U. Bashir and M. Abbas, "A class of quasi-quintic trigonometric Bézier curve with two shape parameters," *Science Asia*, vol. 39, no. 2, pp. 11–15, 2013.
- [16] U. Bashir, M. Abbas, and J. Ali, "The G² and C² rational quadratic trigonometric Bézier curve with two shape parameters with applications," *Applied Mathematics and Computation*, vol. 219, no. 20, pp. 183–197, 2013.
- [17] A. Ghaffar, M. Iqbal, M. Bari et al., "Construction and application of nine-tic B-spline tensor product SS," *Mathematics*, vol. 7, no. 8, p. 675, 2019.
- [18] M. Usman, M. Abbas, and K. T. Miura, "Some engineering applications of new trigonometric cubic Bézier-like curves to free-form complex curve modeling," *Journal of Advanced Mechanical Design Systems and Manufacturing*, vol. 14, no. 4, 2020.
- [19] M. Abbas, N. Ramli, A. A. Majid, and J. M. Ali, "The representation of circular arc by using rational cubic timmer curve," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–6, 2014.
- [20] A. Majeed, M. Abbas, K. T. Miura, M. Kamran, and T. Nazir, "Surface modeling from 2D contours with an application to craniofacial fracture construction," *Mathematics*, vol. 8, no. 8, p. 1246, 2020.
- [21] Y. Yi, L. Hu, C. Liu et al., "Progressive iterative approximation for extended cubic uniform B-splines with shape parameters," *Bulletin of the Malaysian Mathematical Sciences Society*, vol. 10, 2020.
- [22] L. Yan and X. Han, "The extended cubic uniform B-spline curve based on totally positive basis," *Journal of Graphics*, vol. 37, pp. 329–336, 2016.
- [23] H.-W. Lin, H.-J. Bao, and G.-J. Wang, "Totally positive bases and progressive iteration approximation," *Computers & Mathematics with Applications*, vol. 50, no. 3–4, pp. 575–586, 2005.
- [24] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1991.