ARTICLE IN PRESS

Computer-Aided Design ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

The convergence of the geometric interpolation algorithm

Hongwei Lin *

State Key Lab. of CAD&CG, Zhejiang University, Hangzhou, 310027, China

ARTICLE INFO

Article history:
Received 21 July 2009
Accepted 20 January 2010

Keywords:
Geometric interpolation
Convergence
Progressive-iterative approximation
Data fitting
Geometric design

ABSTRACT

The *geometric interpolation algorithm* is proposed by Maekawa et al. in [Maekawa T, Matsumoto Y, Namiki K. Interpolation by geometric algorithm. Computer-Aided Design 2007;39:313–23]. Without solving a system of equations, the algorithm generates a curve (surface) sequence, of which the limit curve (surface) interpolates the given data points. However, the convergence of the algorithm is a conjecture in the reference above, and demonstrated by lots of empirical examples. In this paper, we prove the conjecture given in the reference in theory, that is, the geometric interpolation algorithm is convergent for a blending curve (surface) with normalized totally positive basis, under the condition that the minimal eigenvalue $\lambda_{\min}(D_k)$ of the collocation matrix D_k of the totally positive basis in each iteration satisfies $\lambda_{\min}(D_k) \geq \alpha > 0$. As a consequence, the geometric interpolation algorithm is convergent for Bézier, B-spline, rational Bézier, and NURBS curve (surface) if they satisfy the condition aforementioned, since Bernstein basis and B-spline basis are both normalized totally positive.

1. Introduction

The *geometric interpolation algorithm* is proposed in Ref. [1] by Maekawa et al. for B-spline curve (surface) and subdivision surface fitting (named *interpolation by geometric algorithm* in [1]). Without solving a system of equations, the geometric interpolation algorithm generates a curve (surface) sequence iteratively. Given a data point sequence, an initial curve (surface) is first constructed by taking them as the initial control points. To produce the $(k+1)$th curve (surface) from the kth curve (surface), the geometric interpolation algorithm calculates the foot points on the kth curve (surface) which is the closest to the corresponding data points, constructs the adjusting vectors from the foot points to the data points, and moves the control points of the kth curve (surface) along the adjusting vectors, generating the control points of the $(k+1)$th curve (surface). By this way, a curve (surface) sequence is generated. In Ref. [1], the convergence of the geometric interpolation algorithm is presented as a conjecture, and verified by empirical examples, that is, the limit curve (surface) of the curve (surface) sequence interpolates the given data points.

Although the convergence of the geometric interpolation algorithm is verified only by empirical examples [1], there are some advances in proving the convergence of similar algorithm for subdivision surface fitting recently. In fact, the *progressive interpolation* algorithm is developed for Loop subdivision surface fitting [2,3], Doo–Sabin subdivision fitting [4], and Catmull–Clark subdivision fitting [5]. Its convergence is proved in theory. The progressive interpolation algorithm is the extension of the *progressive-iterative approximation* for blending surfaces [6,7]. However, it can also be regarded as a modification of the geometric interpolation algorithm. The only distinction between the two algorithms lies in the way for calculating the foot points. In progressive interpolation algorithm, the foot points in each iteration are the limit positions of the control points, while in geometric interpolation algorithm, the foot points in kth iteration are the points on the kth surface, which are the closest to the corresponding data points.

In this paper, we prove in theory the convergence of the geometric interpolation algorithm for blending curves and surfaces with normalized totally positive basis, under the condition that the minimal eigenvalue $\lambda_{\min}(D_k)$ of the collocation matrix D_k of the totally positive basis in each iteration satisfies $\lambda_{\min}(D_k) \geq \alpha > 0$. Since Bernstein and B-spline basis are both normalized totally positive, the geometric interpolation algorithm is convergent for Bézier, B-spline, rational Bézier, and NURBS curve and surface, if they fulfill the condition.

This paper is organized as follows. In Section 1.1, the related work is reviewed. Section 2 shows the convergence of the geometric interpolation algorithm for blending curves. Section 3 proves the convergence of the algorithm for blending surfaces. Finally, this paper is concluded in Section 4.

1.1. Related work

Similar to the geometric interpolation algorithm, the progressive-iterative approximation [6,7] is also an iterative method, which generates a curve (surface) sequence. The limit curve (surface) of the sequence interpolates the given data points. Compared

* Tel.: +86 571 88206681 522; fax: +86 571 88206680.
E-mail address: hwlin@zjucadcg.cn.

0010-4485/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2010.01.006

to the geometric interpolation algorithm, where the parameters of foot points in each iteration are changed, the parameters of foot points in each iteration of the progress-iterative approximation are fixed. More clearly, the progressive-iterative approximation depends on the parametric distance (algebraic distance), while the geometric interpolation algorithm depends on the geometric distance.

The progressive-iterative approximation property of the uniform cubic B-spline curve is discovered by Qi [8] and de Boor [9], respectively. In Ref. [6], the authors show that the non-uniform cubic B-spline curve and surface also hold the property. Furthermore, the result is extended to the blending curve and surface with normalized totally positive basis [7]. That is, any blending curve or surface with normalized totally positive basis has the progressive-iterative approximation property. In Ref. [10], the convergence rates of different bases are compared, and the basis with the fastest convergence rate is found. Moreover, it is proved that the rational B-spline curve and surface (NURBS) have the property, too [11]. Recently, Martin et al. [12] devise an iterative format for fitting, which is actually the progressive-iterative approximation format for the uniform periodic cubic B-spline.

Moreover, the progressive-iterative approximation has been extended to subdivision surface fitting [2–5]. And, Lin develops the local progressive-iterative approximation that can fit different data points with different precision requirements [13].

2. Convergence of the curve interpolation algorithm

In Ref. [1], Maekawa et al. develop the *geometric interpolation algorithm*. It is an iterative method, and generates a sequence of curves (surfaces). In Ref. [1], the convergence of the geometric interpolation algorithm is presented as a conjecture, and demonstrated by lots of empirical examples. In this section, we will prove the convergence of the geometric interpolation algorithm to a blending curve with normalized totally positive basis. That is, the limit curve of the curve sequence generated by the geometric interpolation algorithm interpolates the given data points. Since Bernstein basis and B-spline basis are both normalized totally positive basis, the geometric interpolation algorithm is convergent to Bézier, B-spline, rational Bézier, and NURBS curve.

In the following, we first formulate the geometric interpolation algorithm for a blending curve with control points and the normalized totally positive basis $\{B_0(t), B_1(t), \dots, B_n(t)\}$.

Given a data point sequence $\{\boldsymbol{P}_0, \boldsymbol{P}_1, \dots, \boldsymbol{P}_n\}$, an initial blending curve can be constructed as,

$$\boldsymbol{P}^0(t) = \sum_{i=0}^{n} \boldsymbol{P}_i^0 B_i(t), \quad \text{where } \boldsymbol{P}_i^0 = \boldsymbol{P}_i, i = 0, 1, \dots, n. \tag{1}$$

Next, we compute the *foot point* $\boldsymbol{P}^0(t_i^0)$ on the curve $\boldsymbol{P}^0(t)$, which is the closest to the data point $\boldsymbol{P}_i, i = 0, 1, \dots, n$, and construct the *adjusting vector* $\boldsymbol{\Delta}_i^0$ from the foot point to the corresponding data point, that is,

$$\boldsymbol{\Delta}_i^0 = \boldsymbol{P}_i - \boldsymbol{P}^0(t_i^0), \quad i = 0, 1, \dots, n. \tag{2}$$

By adjusting the initial control points $\boldsymbol{P}_i^0$ along the adjusting vectors $\boldsymbol{\Delta}_i^0$, we get the control points $\boldsymbol{P}_i^1, i = 0, 1, \dots, n$ of the next curve,

$$\boldsymbol{P}^1(t) = \sum_{i=0}^{n} \boldsymbol{P}_i^1 B_i(t), \quad \text{where } \boldsymbol{P}_i^1 = \boldsymbol{P}_i^0 + \boldsymbol{\Delta}_i^0, i = 0, 1, \dots, n. \tag{3}$$

Generally, to generate the $(k+1)$th curve $\boldsymbol{P}^{k+1}(t)$ from the kth curve $\boldsymbol{P}^k(t)$, we need to compute the foot point $\boldsymbol{P}^k(t_i^k)$ on the kth curve $\boldsymbol{P}^k(t)$, which is the closest to the data point $\boldsymbol{P}_i$, the adjusting vector $\boldsymbol{\Delta}_i^k = \boldsymbol{P}_i - \boldsymbol{P}^k(t_i^k)$, and the new control point $\boldsymbol{P}_i^{k+1} = \boldsymbol{P}_i^k + \boldsymbol{\Delta}_i^k$, $i = 0, 1, \dots, n$. Thus, the $(k+1)$th curve $\boldsymbol{P}^{k+1}(t)$ is formed as,

$$\boldsymbol{P}^{k+1}(t) = \sum_{i=0}^{n} \boldsymbol{P}_i^{k+1} B_i(t). \tag{4}$$

As a result, we get a curve sequence,

$$\{\boldsymbol{P}^k(t) | k = 0, 1, \dots\}. \tag{5}$$

The convergence of the geometric interpolation algorithm means that, the limit curve interpolates the given data points $\boldsymbol{P}_i, i = 0, 1, \dots, n$. That is,

$$\lim_{k\to\infty} \boldsymbol{P}^k(t_i^k) = \boldsymbol{P}_i, \quad i = 0, 1, \dots, n.$$

To show the convergence of the geometric interpolation algorithm, we introduce a *difference vector* $\bar{\boldsymbol{\Delta}}_i^k$ from the data point $\boldsymbol{P}_i$ to the point $\boldsymbol{P}^k(t_i^{k-1})$ on the kth curve $\boldsymbol{P}^k(t)$, whose parameter t_i^{k-1} is the one of the foot point in the $(k-1)$th iteration, $i = 0, 1, \dots, n$. That is,

$$\begin{aligned}
\bar{\boldsymbol{\Delta}}_i^k &= \boldsymbol{P}_i - \boldsymbol{P}^k(t_i^{k-1}) = \boldsymbol{P}_i - \sum_{j=0}^{n} \boldsymbol{P}_j^k B_j(t_i^{k-1}) \\
&= \boldsymbol{P}_i - \sum_{j=0}^{n} (\boldsymbol{P}_j^{k-1} + \boldsymbol{\Delta}_j^{k-1}) B_j(t_i^{k-1}) \\
&= \boldsymbol{P}_i - \sum_{j=0}^{n} \boldsymbol{P}_j^{k-1} B_j(t_i^{k-1}) - \sum_{j=0}^{n} \boldsymbol{\Delta}_j^{k-1} B_j(t_i^{k-1}) \\
&= \boldsymbol{\Delta}_i^{k-1} - \sum_{j=0}^{n} \boldsymbol{\Delta}_j^{k-1} B_j(t_i^{k-1}), \quad i = 0, 1, \dots, n.
\end{aligned} \tag{6}$$

Then, denote $\bar{\Delta}^k = [\bar{\boldsymbol{\Delta}}_0, \bar{\boldsymbol{\Delta}}_1, \dots, \bar{\boldsymbol{\Delta}}_n]^{\mathrm{T}}$, and $\Delta^k = [\boldsymbol{\Delta}_0, \boldsymbol{\Delta}_1, \dots, \boldsymbol{\Delta}_n]^{\mathrm{T}}$, we have,

$$\bar{\Delta}^k = C_{k-1}\Delta^{k-1} = (I - D_{k-1})\Delta^{k-1}, \tag{7}$$

where, $C_{k-1} = I - D_{k-1}$, I is the identity matrix of rank $(n+1) \times (n+1)$, and,

$$D_{k-1} = \begin{bmatrix} B_0(t_0^{k-1}) & B_1(t_0^{k-1}) & \cdots & B_n(t_0^{k-1}) \\ B_0(t_1^{k-1}) & B_1(t_1^{k-1}) & \cdots & B_n(t_1^{k-1}) \\ \cdots & \cdots & \cdots & \cdots \\ B_0(t_n^{k-1}) & B_1(t_n^{k-1}) & \cdots & B_n(t_n^{k-1}) \end{bmatrix}. \tag{8}$$

It is the collocation matrix of the basis $\{B_i(t), i = 0, 1, \dots, n\}$.

Noticeably, since $\boldsymbol{P}^k(t_i^k)$ is the point on the kth curve $\boldsymbol{P}^k(t)$ closest to the data point $\boldsymbol{P}_i, i = 0, 1, \dots, n$, it is evident that,

$$\left\|\boldsymbol{\Delta}_i^k\right\|_E = \left\|\boldsymbol{P}_i - \boldsymbol{P}^k(t_i^k)\right\|_E \le \left\|\boldsymbol{P}_i - \boldsymbol{P}^k(t_i^{k-1})\right\|_E = \left\|\bar{\boldsymbol{\Delta}}_i^k\right\|_E, \quad i = 0, 1, \dots, n,$$

where, $\|\cdot\|_E$ denotes the Euclidean norm for the vector. Moreover, define

$$\left\|\Delta^k\right\|_M = \max_i \left\{\left\|\boldsymbol{\Delta}_i^k\right\|_E\right\} = \max\left\{\left\|\boldsymbol{\Delta}_0^k\right\|_E, \left\|\boldsymbol{\Delta}_1^k\right\|_E, \dots, \left\|\boldsymbol{\Delta}_n^k\right\|_E\right\},$$

we have

$$\left\|\Delta^k\right\|_M \le \left\|\bar{\Delta}^k\right\|_M, \quad k = 0, 1, \dots. \tag{9}$$

Now, we can present the theorem for the convergence of the geometric interpolation algorithm.

Theorem 1. *If the basis $\{B_i(t), i = 0, 1, \dots, n\}$ is normalized totally positive, $t_0^k < t_1^k < \cdots < t_n^k$, in each iteration $k = 0, 1, \dots$, and the minimal eigenvalue $\lambda_{\min}(D_k)$ of the matrix D_k (8) satisfies $\lambda_{\min}(D_k) \ge \alpha > 0$, the geometric interpolation algorithm is convergent for the blending curve (1), that is, $\lim_{k\to\infty} \boldsymbol{P}^k(t_i^k) = \boldsymbol{P}_i$.*

Proof. Since the basis $\{B_i(t), i = 0, 1, \dots, n\}$ are normalized totally positive basis, and $t_0^k < t_1^k < \cdots < t_n^k$, $k = 0, 1, \dots$, in each iteration, the matrix D_k (8), which is the collocation matrix of the basis $\{B_i(t), i = 0, 1, \ldots, n\}$, is stochastic, and totally positive matrix, whose eigenvalues are all nonnegative real numbers in the interval [0, 1]. Together with $\lambda_{\min}(D_k) \geq \alpha > 0$, and $C_k = I - D_k$, the spectral radius of C_k satisfies $\rho(C_k) = \rho(I - D_k) \leq \beta < 1, k = 0, 1, \ldots$.

On the other hand, since D_k is a totally positive matrix, it can be diagonalized [14,15]. So is $C_k = I - D_k$. That is, there exists an invertible matrix X, such that, $C_k = X^{-1} diag(\lambda_0, \lambda_1, \ldots, \lambda_n)X$, where, $diag(\lambda_0, \ldots, \lambda_n)$ represents the diagonal matrix, and $\lambda_i, i = 0, 1, \ldots, n$ are the eigenvalues of matrix C_k. Thus,

$$\begin{aligned}\|C_k\|_\infty &= \left\|X^{-1} diag(\lambda_0, \ldots, \lambda_n)X\right\|_\infty \\ &\leq \left\|X^{-1}\right\|_\infty \|diag(\lambda_0, \ldots, \lambda_n)\|_\infty \|X\|_\infty \\ &= \|diag(\lambda_0, \ldots, \lambda_n)\|_\infty \leq \rho(C_k) \leq \beta < 1, \quad k = 0, 1, \ldots,\end{aligned}$$

where, $\|\cdot\|_\infty$ is the ∞-norm for the matrix.

In addition, suppose $C_k = [c_{ij}]_{(n+1)\times(n+1)}$, we have,

$$\begin{aligned}\left\|C_k \Delta^k\right\|_M &= \max_i \left\{ \left\| \sum_j c_{ij} \mathbf{\Delta}_j^k \right\|_E \right\} \\ &\triangleq \left\| \sum_j c_{mj} \mathbf{\Delta}_j^k \right\|_E \leq \sum_j |c_{mj}| \left\| \mathbf{\Delta}_j^k \right\|_E \\ &\leq \left(\sum_j |c_{mj}| \right) \max_j \{ \left\| \mathbf{\Delta}_j^k \right\|_E \} \leq \|C_k\|_\infty \left\| \Delta^k \right\|_M .\end{aligned}$$

Therefore, note that $\|\Delta_k\|_M \leq \left\|\bar{\Delta}_k\right\|_M, k = 0, 1, \ldots$. From (9), we have,

$$\begin{aligned}\left\|\Delta^k\right\|_M &\leq \left\|\bar{\Delta}^k\right\|_M = \left\|C_{k-1}\Delta^{k-1}\right\|_M \\ &\leq \|C_{k-1}\|_\infty \left\|\Delta^{k-1}\right\|_M \leq \|C_{k-1}\|_\infty \left\|\bar{\Delta}^{k-1}\right\|_M \\ &= \|C_{k-1}\|_\infty \left\|C_{k-2}\Delta^{k-2}\right\|_M \leq \|C_{k-1}\|_\infty \|C_{k-2}\|_\infty \left\|\Delta^{k-2}\right\|_M \\ &\leq \|C_{k-1}\|_\infty \|C_{k-2}\|_\infty \left\|\bar{\Delta}^{k-2}\right\|_M = \cdots \leq \prod_{i=1}^{k-1} \|C_i\|_\infty \left\|\bar{\Delta}^1\right\|_M \\ &= \prod_{i=1}^{k-1} \|C_i\|_\infty \left\|C_0\Delta^0\right\|_M \leq \prod_{i=0}^{k-1} \|C_i\|_\infty \left\|\Delta^0\right\|_M \leq \beta^k \left\|\Delta^0\right\|_M . \end{aligned} \quad (10)$$

Thus, $\lim_{k\to\infty} \left\|\Delta^k\right\|_M = 0$, meaning that $\lim_{k\to\infty} \boldsymbol{P}^k(t_i^k) = \boldsymbol{P}_i, i = 0, 1, \cdots, n$. □

Now, we are at the position to study the conditions of Theorem 1 in detail. First of all, the condition $t_0^k < t_1^k < \cdots < t_n^k, k = 0, 1, \ldots$ is a reasonable requirement. If it cannot be satisfied, the fitting curve is possible to be self-intersected. Moreover, the fitting precision in the kth iteration will be improved, as long as the minimal eigenvalue of D_k satisfies $\lambda_{\min}(D_k) \geq \alpha > 0$.

As a corollary, since Bernstein basis and B-spline basis are both normalized totally positive bases, if $t_0^k < t_1^k < \cdots < t_n^k$, in each iteration $k = 0, 1, \ldots$, the geometric interpolation algorithm for Bézier curve and B-spline curve are both convergent, under the condition $\lambda_{\min}(D_k) \geq \alpha > 0$. Further, the algorithm for the rational Bézier curve and NURBS curve are also convergent, similar to the progressive-iterative approximation algorithm [11,13].

3. Convergence of the surface interpolation algorithm

In this section, we will show the convergence of the geometric interpolation algorithm for a blending surface with control points and normalized totally positive bases $\{B_0(u), B_1(u), \ldots, B_m(u)\}$ and $\{B_0(v), B_1(v), \ldots, B_n(v)\}$.

Suppose we are given a data point array $\{\boldsymbol{P}_{ij}\}_{i=0\,j=0}^{m\;\;n}$. Taking them as the initial control points, the initial surface $\boldsymbol{S}^0(u, v)$ is constructed as:

$$\boldsymbol{S}^0(u, v) = \sum_{i=0}^m \sum_{j=0}^n \boldsymbol{P}_{ij}^0 B_i(u)B_j(v), \quad \text{where }, \boldsymbol{P}_{ij}^0 = \boldsymbol{P}_{ij}. \quad (11)$$

Similar to the curve case in Section 2, after the kth surface $\boldsymbol{S}^k(u, v) = \sum_{i=0}^m \sum_{j=0}^n \boldsymbol{P}_{ij}^k B_i(u)B_j(v)$ is generated, we calculate the foot point $\boldsymbol{P}^k(u_i^k, v_j^k)$ which is the closest to the data point $\boldsymbol{P}_{ij}$, on the surface $\boldsymbol{S}^k(u, v)$, and the adjusting vector,

$$\mathbf{\Delta}_{ij}^k = \boldsymbol{P}_{ij} - \boldsymbol{S}^k(u_i^k, v_j^k), \quad i = 0, 1, \ldots m, j = 0, 1, \ldots n. \quad (12)$$

By moving the control point $\boldsymbol{P}_{ij}^k$ along the vector $\mathbf{\Delta}_{ij}^k$, we get the control point $\boldsymbol{P}_{ij}^{k+1} = \boldsymbol{P}_{ij}^k + \mathbf{\Delta}_{ij}^k, i = 0, 1, \ldots, m, j = 0, 1, \ldots, n$, for the $(k+1)$th surface, $\boldsymbol{S}^{k+1}(u, v) = \sum_{i=0}^m \sum_{j=0}^n \boldsymbol{P}_{ij}^{k+1} B_i(u)B_j(v)$. Thus, a surface sequence is generated, that is,

$$\{\boldsymbol{S}^k(u, v), k = 0, 1, \ldots, \}. \quad (13)$$

To show the convergence of the geometric interpolation algorithm, namely, $\lim_{k\to\infty} \boldsymbol{S}^k(u_i^k, v_j^k) = \boldsymbol{P}_{ij}$, we need to introduce the difference vector $\bar{\mathbf{\Delta}}_{hl}^k$, from the point $\boldsymbol{S}^k(u_h^{k-1}, v_l^{k-1})$ to the data point $\boldsymbol{P}_{hl}$, whose parameter is the one of the foot point in the $(k-1)$th iteration, $h = 0, 1, \ldots, m, l = 0, 1, \ldots, n$. That is,

$$\begin{aligned}\bar{\mathbf{\Delta}}_{hl}^k &= \boldsymbol{P}_{hl} - \sum_{i=0}^m \sum_{j=0}^n \boldsymbol{P}_{ij}^k B_i(u_h^{k-1})B_j(v_l^{k-1}) \\ &= \boldsymbol{P}_{hl} - \sum_{i=0}^m \sum_{j=0}^n (\boldsymbol{P}_{ij}^{k-1} + \mathbf{\Delta}_{ij}^{k-1}) B_i(u_h^{k-1})B_j(v_l^{k-1}) \\ &= \mathbf{\Delta}_{hl}^{k-1} + \sum_{i=0}^m \sum_{j=0}^n \mathbf{\Delta}_{ij}^{k-1} B_i(u_h^{k-1})B_j(v_l^{k-1}). \end{aligned} \quad (14)$$

Denote,

$$\Delta^k = [\mathbf{\Delta}_{00}^k, \mathbf{\Delta}_{01}^k, \cdots, \mathbf{\Delta}_{0n}^k, \mathbf{\Delta}_{10}^k, \ldots, \mathbf{\Delta}_{1n}^k, \ldots, \mathbf{\Delta}_{m0}^k, \ldots, \mathbf{\Delta}_{mn}^k]^{\mathrm{T}},$$

and,

$$\bar{\Delta}^k = [\bar{\mathbf{\Delta}}_{00}^k, \bar{\mathbf{\Delta}}_{01}^k, \cdots, \bar{\mathbf{\Delta}}_{0n}^k, \bar{\mathbf{\Delta}}_{10}^k, \ldots, \bar{\mathbf{\Delta}}_{1n}^k, \ldots, \bar{\mathbf{\Delta}}_{m0}^k, \ldots, \bar{\mathbf{\Delta}}_{mn}^k]^{\mathrm{T}},$$

we have,

$$\bar{\Delta}^k = F_{k-1}\Delta^{k-1} = (I - G_{k-1})\Delta^{k-1}, \quad (15)$$

where, I is the identity matrix, $G_{k-1} = D_{k-1}^1 \otimes D_{k-1}^2$, with

$$\begin{aligned} D_{k-1}^1 &= \begin{bmatrix} B_0(u_0^{k-1}) & B_1(u_0^{k-1}) & \cdots & B_m(u_0^{k-1}) \\ B_0(u_1^{k-1}) & B_1(u_1^{k-1}) & \cdots & B_m(u_1^{k-1}) \\ \cdots & \cdots & \cdots & \cdots \\ B_0(u_m^{k-1}) & B_1(u_m^{k-1}) & \cdots & B_m(u_m^{k-1}) \end{bmatrix}, \\ D_{k-1}^2 &= \begin{bmatrix} B_0(v_0^{k-1}) & B_1(v_0^{k-1}) & \cdots & B_n(v_0^{k-1}) \\ B_0(v_1^{k-1}) & B_1(v_1^{k-1}) & \cdots & B_n(v_1^{k-1}) \\ \cdots & \cdots & \cdots & \cdots \\ B_0(v_n^{k-1}) & B_1(v_n^{k-1}) & \cdots & B_n(v_n^{k-1}) \end{bmatrix}, \end{aligned} \quad (16)$$

and $\otimes$ denotes the Kronecker product.

Obviously and importantly, since $\boldsymbol{S}^k(u_h^k, v_l^k)$ is the closest to the data point $\boldsymbol{P}_{hl}$ on the kth surface $\boldsymbol{S}^k(u, v)$, we have,

$$\begin{aligned} &\left\|\mathbf{\Delta}_{hl}^k\right\|_E \leq \left\|\bar{\mathbf{\Delta}}_{hl}^k\right\|_E, \quad h = 0, 1, \ldots, m, l = 0, 1, \ldots, n, \text{ and }, \\ &\left\|\Delta^k\right\|_M \leq \left\|\bar{\Delta}^k\right\|_M, \quad k = 0, 1, \ldots. \end{aligned} \quad (17)$$

Similarly, we have the theorem for the convergence of the geometric interpolation algorithm for a blending surface.

Theorem 2. *If the bases $\{B_i(u), i = 0, 1, \ldots, m\}$, and $\{B_j(u), j = 0, 1, \ldots, n\}$ are normalized totally positive, $u_0^k < u_1^k < \cdots < u_m^k$,* *and* $v_0^k < v_1^k < \cdots < v_n^k$ *hold in each iteration* $k = 0, 1, \ldots$, *and the minimal eigenvalues of the matrices* D_k^1 *and* D_k^2 (16) *satisfy* $\lambda_{\min}(D_k^1) \geq \alpha_1 > 0$, *and* $\lambda_{\min}(D_k^2) \geq \alpha_2 > 0$, *respectively, the geometric interpolation algorithm is convergent for the blending surface* (11), *that is*, $\lim_{k\to\infty} \boldsymbol{S}^k(u_i^k, v_j^k) = \boldsymbol{P}_{ij}$.

Proof. The proof of the theorem is similar to that of Theorem 1, and we only give a brief sketch.

First, since $0 < \alpha_1 \leq \lambda_{\min}(D_k^1) < 1$, $0 < \alpha_2 \leq \lambda_{\min}(D_k^2) < 1$, and the eigenvalue of $G_k = D_k^1 \otimes D_k^2$ is the product of the eigenvalues of D_k^1 and D_k^2 [14–16], the minimal eigenvalue of $G_k = D_k^1 \otimes D_k^2$ fulfills $0 < \alpha \leq \lambda_{\min}(G_k) < 1$. Then, the spectral radius $\rho(F_k) = \rho(I - G_k) \leq \beta < 1$.

Second, based on Eqs. (15) and (17), and similar to Eq. (10), we get,

$$\left\|\Delta^k\right\|_M \leq \prod_{i=0}^{k-1} \|F_i\|_\infty \left\|\Delta^0\right\|_M \leq \beta^k \left\|\Delta^0\right\|_M .$$

It leads to $\lim_{k\to\infty} \left\|\Delta^k\right\|_M = 0$, equivalent to $\lim_{k\to\infty} \boldsymbol{S}^k(u_i^k, v_j^k) = \boldsymbol{P}_{ij}$, $i = 0, 1, \ldots, m$, $j = 0, 1, \ldots, n$. □

Specifically, as Bernstein basis and B-spline basis are both normalized totally positive bases, the geometric interpolation algorithm is convergent for Bézier surface, B-spline surface under the condition $0 < \alpha_i \leq \lambda_{\min}(D_k^i) < 1$, $i = 1, 2$, if $u_0^k < u_1^k < \cdots < u_m^k$, and $v_0^k < v_1^k < \cdots < v_n^k$ hold in each iteration. Moreover, it is also convergent for the rational Bézier surface, and NURBS surface, similar to the progressive-iterative approximation [11,13].

4. Conclusion

In this paper, we prove the convergence of the geometric interpolation algorithm for a blending curve (surface) with normalized totally positive basis strictly in theory under the condition $\lambda_{\min}(D_k) \geq \alpha > 0$, which is validated just by empirical examples in Ref. [1]. Specifically, since Bernstein basis and B-spline basis are both normalized totally positive bases, the geometric interpolation algorithm for Bézier, B-spline, rational Bézier, and NURBS curve (surface) is convergent, if they fulfill the condition $\lambda_{\min}(D_k) \geq \alpha > 0$.

Acknowledgements

This paper is supported by Natural Science Foundation of China (Nos. 60970150, 60933008, and 60736019), and Zhejiang Provincial Natural Science Foundation of China (Nos. Y1090416 and R106449).

References

[1] Maekawa T, Matsumoto Y, Namiki K. Interpolation by geometric algorithm. Computer-Aided Design 2007;39:313–23.
[2] Cheng F, Fan F, Lai S, Huang C, Wang J, Yong J. Progressive interpolation using loop subdivision surfaces. In: Chen F, Jüttler, editors. Proceedings of geometric modeling and processing (GMP). LNCS, vol. 4975. 2008. p. 526–33.
[3] Cheng F, Fan F, Lai S, Huang C, Wang J, Yong J. Loop subdivision surface based progressive interpolation. Journal of Computer Science and Technology 2009; 24(1):39–46.
[4] Fan F, Cheng F, Lai S. Subdivision based interpolation with shape control. Computer Aided Design & Applications 2008;5(1–4):539–47.
[5] Chen Z, Luo X, Tan L, Ye B, Chen J. Progressive interpolation based on Catmull-Clark subdivision surfaces. Pacific Graphic 2008. Computer Grahics Forum 2008;27(7):1823–7.
[6] Lin H, Wang G, Dong C. Constructing iterative non-uniform B-spline curve and surface to fit data points. SCIENCE IN CHINA, Series F 2004;47(3):315–31.
[7] Lin H, Bao H, Wang G. Totally positive bases and progressive iteration approximation. Computers and Mathematics with Applications 2005;50(3–4): 575–86.
[8] Qi D, Tian Z, Zhang Y, Feng JB. The method of numeric polish in curve fitting. Acta Mathematica Sinica 1975;18(3):173–84.
[9] de Boor C. How does Agee's smoothing method work? In: Proceedings of the 1979 Army Numerical Analysis and Computers Conference, ARO Report 79-3, Army Research Office. 1979. p. 299-302.
[10] Delgado J, Peña JM. Progressive iterative approximation and bases with the fastest convergence rates. Computer Aided Geometric Design 2007;24(1): 10–8.
[11] Shi L, Wang R. An iterative algorithm of NURBS interpolation and approximation. Journal of Mathematical Research and Exposition. 2006;26(4): 735–43.
[12] Martin T, Cohen E, Kirby RM. Volumetric parameterization and trivariate B-spline fitting using harmonic functions. Computer Aided Geometric Design 2009;26(6):648–64.
[13] Lin H. Local progressive-iterative approximation format for blending curves and patches. Computer Aided Geometric Design, in press [doi:10.1016/j.cagd.2010.01.003].
[14] Ando T. Totally positive matrices. Linear Algebra and its Applications 1987;90: 165–219.
[15] Karlin S. Total positivity, Volumn I. Standord, C.A: Standford University Press; 1968.
[16] Chen JL, Chen XH. Special matrices. Beijing: Tsinghua University Press; 2000. [in Chinese].