



Research article

Hermitian and skew-Hermitian splitting method on a progressive-iterative approximation for least squares fitting

Saknarin Channark^{1,2}, Poom Kumam^{1,2,3,*}, Juan Martinez-Moreno⁴ and Wachirapong Jirakitpuwapat⁵

¹ Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road, Bang Mod, Thung Khru, Bangkok 10140, Thailand

² Center of Excellence in Theoretical and Computational Science (TaCS-CoE), Science Laboratory Building, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road, Bang Mod, Thung Khru, Bangkok 10140, Thailand

³ Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 40402, Taiwan

⁴ Faculty of Experimental Science, Department of Mathematics, University of Jaen, Campus Las Lagunillas, s/n, 23071, Jaen, Spain

⁵ National Security and Dual-Use Technology Center, National Science and Technology Development Agency (NSTDA), Phatum Thani 12120, Thailand

* **Correspondence:** Email: poom.kum@kmutt.ac.th.

Abstract: To solve the problems of curves and surfaces approximation with normalized totally positive bases, a new progressive and iterative approximation for least square fitting method called HSS-LSPIA is proposed, which is based on the HSS iterative approach for solving linear equations of LSPIA. The HSS-LSPIA format includes two iterations with iterative difference vectors, each of which is distinct from the other. The approximate optimal positive constant, as well as convergence analyses, are provided. Furthermore, the HSS-LSPIA method can be faster than the ELSPIA, LSPIA, and WHPIA methods in terms of convergence speed. Numerical results verify this phenomenon.

Keywords: progressive-iterative approximation; HSS iteration; least square fitting

Mathematics Subject Classification: 41A15, 65D10, 65D17, 65F10, 65F15

1. Introduction

Geometric iteration techniques (GI) are a subcategory of iterative linear algebraic methods for solving linear equations that have gained popularity as a data-fitting tool in recent years. Since its

introduction, geometric iteration approaches in geometric design, including computer-aided design (CAD), differential equations, statistics, and approximation theory, have been widely used in many areas of research and practical applications. The progressive and iterative approximation (PIA) method presented by Lin et al. [1] is an adaptable and stable approach to data fitting that belongs to the geometric iteration methods category. The PIA method can be applied to interpolate and approximate curves and surfaces with normalized totally positive (NTP) bases.

Moreover, the EPIA method [2] is extended to approximate a given data points set. The EPIA organizes the data points into groups with equal to the amount of control points as there are control points. Each group represents a control point and then executes PIA-format iterative approximation. However, the limit of the generated curves and surfaces sequence is not the least square fitting (LSF) result to the data points set. Then, Deng and Lin proposed the LSPIA method [3], in which the least square fitting (LSF) result to the data set with the B-spline basis is the limit of the generated curves and surfaces sequence. Ebrahimi and Loghman [4] presented a composite iterative procedure for the progressive and iteration approximation for generates a sequence of matrices based on the Schulz iterative method which it uses in the adjusting vectors. In addition, Lin et al. [5] also gives an review of interpolation and approximation geometric iteration methods, as well as local characteristics and accelerating methods, as well as demonstrating convergence.

Recently, Wang [6] presented an extended the progressive and iterative approximation for least square fitting (ELSPIA) approach with global and local relaxation parameters, which converges faster than the LSPIA method [3] in terms of iterative steps and computing time. In this article, we propose the Hermitian and skew-hermitian splitting methods for least squares fitting on a progressive-iterative approximation, can theoretically reduce the approximation error to zero.

The PIA [1] is an iterative approach for solving a system of linear equations $BP = Q$, as we know, where $B = \{B_i(t_j)\}_{i,j=1}^{n,m}$ is a collocation matrix with $0 = t_1 < t_2 < t_3 < \dots < t_m = 1$ and $n = m$, $\{Q_j\}_{j=1}^m$ refers to a collection of data points and $\{P_i\}_{i=1}^n$ refers to a collection of control points. When all of the entries in each row of the blending basis matrix B add up to 1, B is normalized. Consider that if all of a matrix's minors are non-negative, it is totally positive. The basis is considered to be totally positive if all of the collocation matrices are totally positive. As a result, the blending basis matrix B is normalized totally positive (NTP), and the NTP basis is the corresponding basis. The NTP bases have traditionally been related to shape-preserving representations [7]. The majority of curve and surface representations used in computer-aided design, such as the Bernstein basis and the B-spline basis, are based on NTP bases.

Using the Hermitian and skew-Hermitian splits, Bai et al. [8] presented an iteration approach (HSS). Hu et al. [9] developed a new PIA iterative algorithm and its weighted form for progressive iterative interpolation of data points based on the HSS-iteration approach, using NTP bases, namely HPIA and WHPIA, respectively. These two approaches' iterative processes formally consist of two iterations, with the iterative distinction vectors in the two iterations differing.

As we know, the LSPIA [3, 10] and ELSPIA [6] are the iteration approaches for solving a system of linear equations $\mu B^T B P = B^T Q$, where $B = \{B_i(t_j)\}_{i,j=1}^{n,m}$ is a collocation matrix with $0 = t_1 < t_2 < t_3 < \dots < t_m = 1$ and $m \geq n$, μ is constant satisfying the condition $0 < \mu < \frac{2}{\lambda_0}$ with λ_0 is the largest eigenvalue of $B^T B$, $\{Q_j\}_{j=1}^m$ refers to a collection of data points and $\{P_i\}_{i=1}^n$ refers to control points.

According to theoretical analysis [8], the HSS-iteration consistently converges unconditionally to the exact solution of the system of linear equations $Ax = b$, where $A \in \mathbb{C}^{n \times n}$ and non-singular. Using

the HSS-iteration approach, we propose a new procedure, called the HSS-LSPIA method, for solving a system of linear equations of LSPIA such that $A = \mu B^T B$, $x = P$ and $b = B^T Q$, where $B = \{B_i(t_j)\}_{i,j=1}^{n,m} \in \mathbb{R}^{m \times n}$ is a collocation matrix with $m \geq n$.

The paper is organized as the following: Section 2 presents preliminaries. Section 3 shows the iterative format of the HSS-LSPIA method schemes for curves and surfaces with NTP bases respectively. And proves the convergence of HSS-LSPIA in the case of curves and surfaces in this section. In Section 4, we present the experimental results of five examples show that the HSS-LSPIA method is more effectively efficient than the LSPIA [3], ELSPIA [6] and WHPIA [9]. Finally, in Section 5, some conclusions are summarized.

2. Preliminaries

2.1. LSPIA and ELSPIA

For any blending curves and surfaces fitting, we'll go through the progressive iterative approximation property. For the ordered point set $\{Q_j\}_{j=1}^m$, select control point set $\{P_i\}_{i=1}^n$ from $\{Q_j\}_{j=1}^m$ with $m \geq n$, let $\{Q_j\}_{j=1}^m$ be the assigned parameters to $\{Q_j\}_{j=1}^m$, choose $B = \{B_i(t_j)\}_{i,j=1}^{n,m}$ as an NTP basis. For the ELSPIA method [6], let $\gamma_j^0 = 0$, $j = 1, 2, \dots, m$, and $P_i^0 = P_i$, $i = 1, 2, \dots, n$ be weighted sum and the initial control points, respectively, we can obtain the $(k + 1)$ th blending curve,

$$C^{k+1}(t) = \sum_{i=1}^n B_i(t) P_i^{k+1}, \quad t \in [0, 1]. \quad (2.1)$$

We update the control points

$$P_i^{k+1} = P_i^k + \Delta_i^{k+1}, \quad i = 1, 2, \dots, n, \quad (2.2)$$

by adjusting vector

$$\Delta_i^{k+1} = \mu \sum_{j=1}^m B_i(t_j) \gamma_j^{k+1}, \quad i = 1, 2, \dots, n, \quad (2.3)$$

with the inner iteration

$$\gamma_j^{k+1} = (1 - \omega) \gamma_j^k + \omega \delta_j^k, \quad j = 1, 2, \dots, m, \quad (2.4)$$

and the difference vector

$$\delta_j^k = Q_j - \sum_{i=1}^n B_i(t_j) P_i^k, \quad j = 1, 2, \dots, m. \quad (2.5)$$

The parameters ω and μ both satisfy the condition in this case

$$0 < \omega < 2, \quad 0 < \mu < \frac{2(2 - \omega)}{\omega \sigma_0^2} \quad (2.6)$$

where σ_0 stands for the collocation matrix B 's largest singular value.

When the parameters ω and μ are satisfy as to the following

$$\omega = 1, \quad 0 < \mu < \frac{2}{\sigma_0^2} \quad (2.7)$$

the LSPIA method [3] is a special case of the ELSPIA method for blending curve fitting.

In the case of the surface, two NTP bases are required. In both directions t and v , a Kronecker product is formed by $\{B_i(t)\}_{i=1}^{n_1}$ and $\{N_j(v)\}_{j=1}^{n_2}$. We given the initial control points $\{P_{ij}^0\}_{i,j=1}^{n_1, n_2}$, with a real increasing parameters set $\{t_h^0\}_{h=1}^{m_1}$, i.e. $0 = t_1 < t_2 < t_3 < \dots < t_{m_1} = 1$ and $\{v_l^0\}_{l=1}^{m_2}$, i.e. $0 = v_1 < v_2 < v_3 < \dots < v_{m_2} = 1$. We select $\{P_{ij}\}_{i,j=1}^{n_1, n_2}$ from the ordered point set $\{Q_{ij}\}_{i,j=1}^{m_1, m_2}$. The sequence of surface iterations $C^{(k+1)}(t, v)$ is related to that of the curve case with $k \geq 0$;

$$C^{k+1}(t, v) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t)N_j(v)P_{i,j}^{k+1}. \quad (2.8)$$

We update the control points

$$P_{ij}^{k+1} = P_{ij}^k + \Delta_{ij}^{k+1}, \quad i = 1, 2, \dots, n_1, \quad j = 1, 2, \dots, n_2, \quad (2.9)$$

by adjusting vector

$$\Delta_{ij}^{k+1} = \mu \sum_{h=1}^{m_1} \sum_{l=1}^{m_2} B_i(t_h)N_j(v_l)\gamma_{hl}^{k+1}, \quad i = 1, 2, \dots, n_1 \text{ and } j = 1, 2, \dots, n_2, \quad (2.10)$$

with the inner iteration

$$\gamma_{ij}^{k+1} = (1 - \omega)\gamma_{ij}^k + \omega\delta_{ij}^k, \quad i = 1, 2, \dots, m_1, \quad j = 1, 2, \dots, m_2, \quad (2.11)$$

and the difference vector

$$\delta_{ij}^k = Q_{ij} - \sum_{h=1}^{m_1} \sum_{l=1}^{m_2} B_i(t_h)N_j(v_l)\gamma_{hl}^k, \quad (2.12)$$

with $i = 1, 2, \dots, m_1$ and $j = 1, 2, \dots, m_2$, where μ is the global relaxation parameter and ω is the local relaxation parameter satisfy

$$0 < \omega < 2, \quad 0 < \mu < \frac{2(2 - \omega)}{\omega\sigma_{B_0}^2\sigma_{N_0}^2} \quad (2.13)$$

where σ_{B_0} and σ_{N_0} are the largest singular values of the collocation matrix B and N, which correspond to the bases $\{B_i(t)\}_{i=1}^{n_1}$ and $\{N_j(v)\}_{j=1}^{n_2}$, respectively.

When the parameters ω and μ are satisfy as to the following

$$\omega = 1, 0 < \mu < \frac{2(2 - \omega)}{\omega\sigma_{B_0}^2\sigma_{N_0}^2}. \quad (2.14)$$

the LSPIA method [3] is a special case of the ELSPIA method for blending surface fitting.

Let

$$\begin{aligned} \Gamma^{k+1} &= [\gamma_0^{k+1}, \gamma_1^{k+1}, \dots, \gamma_m^{k+1}]^T, \quad k \geq 0, \\ P^{k+1} &= [P_0^{k+1}, P_1^{k+1}, \dots, P_n^{k+1}]^T, \quad k \geq 0, \end{aligned}$$

and

$$Q = [Q_0, Q_1, \dots, Q_n]^T.$$

According to updating the control points in the case of curve (2.2), we revise the iterative form with (2.3)–(2.5) as the matrix iterative form in the same matrix form as updating the control points in the case of surface (2.9). We then rewrite with (2.10)–(2.12) as follows

$$\begin{cases} \Gamma^{k+1} = (1 - \omega)\Gamma^k + \omega(Q - B P^k), \\ P^{k+1} = P^k + \mu B^T \Gamma^{k+1}, \quad k \geq 0. \end{cases} \quad (2.15)$$

2.2. HSS-iteration

Many problems in scientific computing lead to a system of linear equations

$$BP = Q, \quad B \in \mathbb{C}^{m \times m} \text{ is nonsingular, } P, Q \in \mathbb{C}^m, \quad (2.16)$$

with B a large sparse non-Hermitian and positive definite matrix.

Iterative methods for the system of linear equations (2.16) require efficient splittings of the coefficient matrix B . For example, the Jacobi and the Gauss-Seidel iterations [11–16] split the matrix B into its diagonal and off-diagonal part. Because the matrix B naturally possesses a Hermitian/skew-Hermitian splitting (HSS) [17, 18]

$$B = H + S, \quad (2.17)$$

where

$$H = \frac{1}{2}(B + B^T) \quad \text{and} \quad S = \frac{1}{2}(B - B^T), \quad (2.18)$$

Bai et al. presented the Hermitian/skew-hermitian splitting (HSS) iteration approach [9] as a new way to solve a system of linear equations (2.16). We'll pretend that we've received the iteration sequence until P^{k+1} ($k \geq 1$) converges. Then, the HSS iteration can compute from the format as

$$\begin{cases} (\alpha I + H)P^{k+\frac{1}{2}} = (\alpha I - S)P^k + Q, \\ (\alpha I + S)P^{k+1} = (\alpha I - H)P^{k+\frac{1}{2}} + Q, \end{cases} \quad (2.19)$$

where I is the identity matrix and α is a positive constant.

They have also proved this method converges unconditionally to the exact solution of the system of linear equations (2.16). The upper bound of the contraction factor of the HSS iteration is dependent on the spectrum of the Hermitian part H but is independent of the spectrum of the skew-Hermitian part S as well as the eigenvectors of the matrices H, S , and B .

2.3. WHPIA iteration

Hu et al. [9] presented a new PIA iterative technique and its weighted edition for progressive iterative interpolation of data points based on the HSS-iteration approach, using NTP bases called HPIA and WHPIA, respectively. The WHPIA method typically converges faster than the HPIA method in the majority of cases. Following that, we briefly review the WHPIA' iterative processes, which formally consist of two iterations with different iterative distinction vectors in the two iterations, and set up a function based on NTP bases like a perturbation term in the iteration process.

The case of a curve with an NTP basis $\{B_i(t)\}_{i=1}^m$ and a collection of data points to be interpolated $\{Q_i\}_{i=1}^m$. Then, with a real increasing parameters set $\{t_i\}_{i=1}^m$ i.e. $0 = t_1 < t_2 < t_3 < \dots < t_m = 1$, use these data points to be the initial control points $\{P_i^0\}_{i=1}^m$ and $\{P_i^{\frac{1}{2}}\}_{i=1}^m$. The initial blending curve can be formed as follows:

$$C^0(t) = \sum_{i=1}^m B_i(t)P_i^0, \quad t \in [0, 1], \quad (2.20)$$

and

$$C^{\frac{1}{2}}(t) = \sum_{i=1}^m B_i(t)P_i^{\frac{1}{2}}, \quad t \in [0, 1]. \quad (2.21)$$

Then, the sequence of curves $C^k(t)$ for $k \geq 1$ as the following

$$C^k(t) = \sum_{i=1}^m B_i(t)P_i^k, \quad t \in [0, 1], \quad (2.22)$$

where

$$\begin{cases} P_i^k &= P_i^{k-\frac{1}{2}} + \omega\Delta_{1i}^k, \\ \Delta_{1i}^k &= P_i^0 - \left[\frac{1}{2}(C^k(t_i) + C^{k-\frac{1}{2}}(t_i)) + \frac{1}{2}(C^{k-\frac{1}{2}}(B_i) + C^k(B_i)) \right], \\ P_i^{k+\frac{1}{2}} &= P_i^k + \omega\Delta_{2i}^k, \\ \Delta_{2i}^k &= P_i^0 - \left[\frac{1}{2}(C^{k-1}(t_i) + C^{k-\frac{1}{2}}(t_i)) + \frac{1}{2}(C^{k-\frac{1}{2}}(B_i) + C^{k-1}(B_i)) \right]. \end{cases} \quad (2.23)$$

In (2.23), $C^k(B_i)$ is defined as follows

$$C^k(B_i) = \sum_{j=1}^m B_i(t_j)P_j^k, \quad i = 1, 2, \dots, m, \quad k = 0, \frac{1}{2}, 1, \dots,$$

which is a function that receives the blending bases as variables.

Surface fitting with two NTP bases $\{B_i(t)\}_{i=1}^{n_1}$, $\{N_j(v)\}_{j=1}^{n_2}$, and a collection of data points $\{Q_{ij}\}_{i,j=1}^{n_1, n_2}$ to be interpolated. Then, using these data points to be the initial control points $\{P_{ij}^0\}_{i,j=1}^{n_1, n_2}$, with a real increasing parameters set $\{t_i^0\}_{i=1}^{n_1}$, i.e. $0 = t_1 < t_2 < t_3 < \dots < t_{n_1} = 1$ and $\{v_j^0\}_{j=1}^{n_2}$, i.e. $0 = v_1 < v_2 < v_3 < \dots < v_{n_2} = 1$. The initial blending surface can be created as follows:

$$C^0(t, v) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t)N_j(v)P_{i,j}^0, \quad t, v \in [0, 1]. \quad (2.24)$$

Then, the sequence of surfaces $C^{k+1}(t, v)$ for $k \geq 1$, can be compute as follows

$$C^k(t, v) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t)N_j(v)P_{i,j}^k, \quad (2.25)$$

where

$$\begin{cases} P_{ij}^k &= P_{ij}^{k-\frac{1}{2}} + \omega\Delta_{1ij}^k, \\ \Delta_{1ij}^k &= P_{ij}^0 - \left[\frac{1}{2}(C^k(t_i, v_j) + C^{k-\frac{1}{2}}(t_i, v_j)) + \frac{1}{2}(C^{k-\frac{1}{2}}(B_i, N_j) + C^k(B_i, N_j)) \right], \\ P_{ij}^{k+\frac{1}{2}} &= P_{ij}^k + \omega\Delta_{2ij}^k, \\ \Delta_{2ij}^k &= P_{ij}^0 - \left[\frac{1}{2}(C^{k-1}(t_i, v_j) + C^{k-\frac{1}{2}}(t_i, v_j)) + \frac{1}{2}(C^{k-\frac{1}{2}}(B_i, N_j) + C^{k-1}(B_i, N_j)) \right]. \end{cases} \quad (2.26)$$

We can use the following expression in (2.26) as a perturbation term in the iteration process, even as we appear to have done with a curve;

$$C^k(B_i, N_j) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t)N_j(v)P_{i,j}^k, \quad k = 0, \frac{1}{2}, 1, \dots$$

The WHPIA format of curves and surface, which involves two iterations and substituting the sequential iterative process length of each control point in PIA, is referred to as (2.23) and (2.26), respectively.

3. Iterative format of HSS-LSPIA

As we know, the PIA [1] is an iterative approach for solving a system of linear equations $BP = Q$, $B \in \mathbb{C}^{m \times n}$ with $n = m$. The matrix form of PIA method as

$$\begin{aligned} P^{k+1} &= P^k + (Q - BP^k) \\ &= (I - B)P^k + Q, \end{aligned} \quad (3.1)$$

and the LSPIA [3, 10] and ELSPIA [6] are the iteration approaches for solving a system of linear equations $\mu B^T BP = B^T Q$, $B \in \mathbb{C}^{m \times n}$ with $m \geq n$. The matrix form of LSPIA method as

$$\begin{aligned} P^{k+1} &= P^k + \mu B^T (Q - BP^k) \\ &= (I - \mu B^T B)P^k + \mu B^T Q. \end{aligned} \quad (3.2)$$

For an NTP basis $\{B_i(t_j)\}_{i,j=1}^{n,m}$ and a data set points $\{Q_j\}_{j=1}^m$ to be fitted in either $\mathbb{R}^{m \times 2}$ or $\mathbb{R}^{m \times 3}$. We use all these data points to be the initial control points $\{P_i^0\}_{i=1}^n$ in $\mathbb{R}^{n \times 2}$ or $\mathbb{R}^{n \times 3}$ and combine the HSS-iteration with the LSPIA method for solving a system of linear equations

$$\mu B^T BP = B^T Q, \quad B \in \mathbb{C}^{m \times n} \quad (m \geq n). \quad (3.3)$$

The matrix form of HSS-LSPIA method as the following

$$\begin{cases} (\alpha I + H)P^{k+\frac{1}{2}} = (\alpha I - S)P^k + \mu B^T Q, \\ (\alpha I + S)P^{k+1} = (\alpha I - H)P^{k+\frac{1}{2}} + \mu B^T Q, \end{cases} \quad (3.4)$$

where

$$\begin{aligned} H &= \frac{\mu}{2} (B^T B + (B^T B)^T) \\ &= \frac{\mu}{2} (B^T B + B^T B) = \mu B^T B \end{aligned} \quad (3.5)$$

and

$$\begin{aligned} S &= \frac{\mu}{2} (B^T B - (B^T B)^T) \\ &= \frac{\mu}{2} (B^T B - B^T B) = 0. \end{aligned} \quad (3.6)$$

We can rewrite (3.4) as

$$\begin{cases} (\alpha I + \mu B^T B)P^{k+\frac{1}{2}} = (\alpha I)P^k + \mu B^T Q, \\ (\alpha I)P^{k+1} = (\alpha I - \mu B^T B)P^{k+\frac{1}{2}} + \mu B^T Q. \end{cases} \quad (3.7)$$

Then, we have

$$\alpha IP^{k+\frac{1}{2}} + \mu B^T BP^{k+\frac{1}{2}} = \alpha IP^k + \mu B^T Q, \quad (3.8)$$

and

$$\alpha IP^{k+1} = \alpha IP^{k+\frac{1}{2}} - \mu B^T BP^{k+\frac{1}{2}} + \mu B^T Q. \quad (3.9)$$

By (3.8) and (3.9), we get

$$\alpha IP^{k+1} = \alpha IP^k + \mu B^T Q - \mu B^T BP^{k+\frac{1}{2}} + \mu B^T Q - \mu B^T BP^{k+\frac{1}{2}}. \quad (3.10)$$

So,

$$P^{k+1} = P^k + \frac{\mu}{\alpha} B^T (Q - BP^{k+\frac{1}{2}}) + \frac{\mu}{\alpha} B^T (Q - BP^{k+\frac{1}{2}}). \quad (3.11)$$

Let

$$P_i^{k+\frac{1}{2}} = P^k + \frac{\mu}{\alpha} B^T (Q - BP^{k+\frac{1}{2}})$$

and

$$\Delta_i^k = \frac{\mu}{\alpha} B^T (Q - BP^{k+\frac{1}{2}}),$$

then

$$P_i^{k+1} = P_i^{k+\frac{1}{2}} + \Delta_i^k. \quad (3.12)$$

Remark 3.1. Since the skew-Hermitian part $S = 0$, HSS-LSPIA can called shifted splitting LSPIA [19].

3.1. The curve case

Considering an NTP basis $\{B_i(t_j)\}_{i,j=1}^{n,m}$ with $m \geq n$ and a data set points $\{Q_j\}_{j=1}^m$ to be fitted in either $\mathbb{R}^{m \times 2}$ or $\mathbb{R}^{m \times 3}$. The initial blending curve $C^0(t)$ and $C^{\frac{1}{2}}(t)$ can then be generated using all these data points to be the initial control points $\{P_i^0\}_{i=1}^n$ in $\mathbb{R}^{n \times 2}$ or $\mathbb{R}^{n \times 3}$ and $\{P_i^{\frac{1}{2}}\}_{i=1}^n$, with a real increasing parameters set $\{t_i\}_{i=1}^m$ i.e. $0 = t_1 < t_2 < t_3 < \dots < t_m = 1$. The initial blending curve can be formed as follows:

$$C^0(t) = \sum_{i=1}^n B_i(t) P_i^0, \quad t \in [0, 1] \quad (3.13)$$

and

$$C^{\frac{1}{2}}(t) = \sum_{i=1}^n B_i(t) P_i^{\frac{1}{2}}, \quad t \in [0, 1]. \quad (3.14)$$

Then, the remaining curve of the sequence $C^{k+1}(t)$ for $k \geq 0$, can be compute as following:

$$C^{k+1}(t) = \sum_{i=1}^n B_i(t) P_i^{k+1}, \quad t \in [0, 1], \quad (3.15)$$

where

$$\begin{cases} P_i^{k+1} &= P_i^{k+\frac{1}{2}} + \Delta_i^k, \\ \Delta_i^k &= \frac{\mu}{\alpha} B^T (Q_i - C^{k+\frac{1}{2}}(t_i)), \\ P_i^{k+\frac{1}{2}} &= P_i^k + \frac{\mu}{\alpha} B^T (Q_i - C^{k+\frac{1}{2}}(t_i)). \end{cases} \quad (3.16)$$

In (3.16), $C^{k+\frac{1}{2}}(t_j)$ is defined as the following:

$$C^{k+\frac{1}{2}}(t_j) = \sum_{i=1}^n B_i(t_j) P_i^{k+\frac{1}{2}}, \quad k = 0, 1, 2, \dots \quad (3.17)$$

Both global and local relaxation parameters, μ and α are satisfied as

$$0 < \mu < \frac{2}{\lambda_{max}}, \quad \alpha > 0 \quad (3.18)$$

where λ_{max} denotes the largest eigenvalue of $B^T B$. If (3.16) converge, then

$$\lim_{k \rightarrow \infty} P_i^k = \lim_{k \rightarrow \infty} P_i^{k+\frac{1}{2}} = \lim_{k \rightarrow \infty} P_i^{k+1}, \quad (3.19)$$

the HSS-LSPIA format of curves, including of two iterations and substituting the sequential iterative process from each control point in LSPIA, is recognized as (3.16).

Remark 3.2. From (3.16), we will get $\lim_{k \rightarrow \infty} C^{k+1}(t_i)$ exist, if for any $\epsilon > 0$ there is a natural number A , when $k > A$, $\|P_i^{k+1} - P_i^k\| < \epsilon$. Furthermore, whether the spectral radius of the iterative matrix P^{k+1} in (3.11) is less than 1 affects convergence.

3.2. Convergence analysis of the HSS-LSPIA curve fitting

Lemma 3.3. [1] Given an NTP basis $\{B_i(t_j)\}_{i,j=1}^{n,m}$ with $m \geq n$ and $B^T B$ is nonsingular collocation matrix. And assuming that $\lambda_i(B^T B)$, $i = 1, 2, \dots, n$ is eigenvalue of $B^T B$, then $0 < \lambda_i(B^T B) \leq 1$.

Let $B \in \mathbb{C}^{m \times n}$, $B = M_i - N_i$ ($i = 1, 2$) be two splitting of the matrix B . In the matrix form, Eq (3.16)'s two iterative processes can be written as

$$\begin{cases} M_1 P^{k+\frac{1}{2}} = N_1 P^k + \mu B^T Q, \\ M_2 P^{k+1} = N_2 P^{k+\frac{1}{2}} + \mu B^T Q, \end{cases} \quad (3.20)$$

where $M_1 = (\alpha I + \mu B^T B)$, $N_1 = \alpha I$, $M_2 = \alpha I$, $N_2 = (\alpha I - \mu B^T B)$, I is identity matrix. Then we can rewrite (3.20) as a unified matrix iteration format as follows

$$P^{k+1} = M_2^{-1} N_2 M_1^{-1} N_1 P^k + \mu M_2^{-1} (I + N_2 M_1^{-1}) B^T Q, \quad k = 0, 1, 2, \dots \quad (3.21)$$

or

$$P^{k+1} = M P^k + \mu M_0 B^T Q, \quad k = 0, 1, 2, \dots \quad (3.22)$$

where $M = M_2^{-1} N_2 M_1^{-1} N_1$ and $M_0 = M_2^{-1} (I + N_2 M_1^{-1})$.

Now we have to show that the iterative control point sequence $\{P^{k+1}\}$ leads to the unique solution P^* , i.e. $\rho(M) < 1$.

Theorem 3.4. Let $B \in \mathbb{C}^{m \times n}$ with $m \geq n$ be a positive definite matrix, let $H = \frac{\mu}{2} (B^T B + (B^T B)) = \mu B^T B$ and $S = \frac{\mu}{2} (B^T B - (B^T B)) = 0$ be its Hermitian and skew-Hermitian part, and given α be a positive constant. The HHS-LSPIA iteration's iteration matrix $M(\alpha)$ is then provided by

$$M(\alpha) = (\alpha I + S)^{-1} (\alpha I - H) (\alpha I + H)^{-1} (\alpha I - S)$$

or

$$M(\alpha) = (\alpha I)^{-1} (\alpha I - \mu B^T B) (\alpha I + \mu B^T B)^{-1} (\alpha I) \quad (3.23)$$

and the spectral radius of $\hat{M}(\alpha)$ is $\rho(\hat{M}(\alpha))$, bounded by

$$\sigma(\alpha) \equiv \max_{\lambda_i \in \lambda(H)} \left| \frac{\alpha - \lambda_i}{\alpha + \lambda_i} \right|,$$

where the spectral set of the matrix H is $\lambda(H)$.

As a consequence, it maintains that

$$\rho(M(\alpha)) \leq \sigma(\alpha) < 1 \quad \forall \alpha > 0,$$

i.e., the HHS-LSPIA iteration converges to the unique solution P^* of the linear equation (3.3).

Proof. By the resemblance invariance to the matrix spectrum, we have

$$\begin{aligned} \rho(M(\alpha)) &= \rho((\alpha I)^{-1}(\alpha I - \mu B^T B)(\alpha I + \mu B^T B)^{-1}(\alpha I)) \\ &= \rho((\alpha I - \mu B^T B)(\alpha I + \mu B^T B)^{-1}(\alpha I)(\alpha I)^{-1}) \\ &\leq \|(\alpha I - \mu B^T B)(\alpha I + \mu B^T B)^{-1}(\alpha I)(\alpha I)^{-1}\|_2 \\ &\leq \|(\alpha I - \mu B^T B)(\alpha I + \mu B^T B)^{-1}\|_2 \cdot \|(\alpha I)(\alpha I)^{-1}\|_2. \end{aligned}$$

We know that $(\alpha I)(\alpha I)^{-1} = I$ which is unitary matrix i.e., $\|(\alpha I)(\alpha I)^{-1}\|_2 = 1$. Then, we get

$$\rho(M(\alpha)) < \|(\alpha I - \mu B^T B)(\alpha I + \mu B^T B)^{-1}\|_2 = \max_{\lambda_i \in \lambda(H)} \left| \frac{\alpha - \lambda_i}{\alpha + \lambda_i} \right|.$$

Since $\lambda_i > 0$ ($i = 1, 2, 3, \dots, n$) and α is positive constant, it is easy to see that $\rho(M(\alpha)) \leq \rho(\alpha) < 1$. \square

Corollary 3.5. Under the same assumption of Theorem 3.4. Let λ_{\max} and λ_{\min} be the maximum and the minimum eigenvalues of $B^T B$, respectively. Then, we have the approximate fastest convergence rate of HHS-LSPIA method for the curve fitting when

$$\alpha^* = \mu \sqrt{\lambda_{\max}(B^T B)\lambda_{\min}(B^T B)} \quad (3.24)$$

where $\mu = \frac{2}{\lambda_{\max} + \lambda_{\min}}$ presented in [3].

Proof. In [8], it is shown that the optimal spectral radius. Now,

$$\sigma(\alpha) \equiv \max \left\{ \left| \frac{\alpha - \lambda_{\min}}{\alpha + \lambda_{\min}} \right|, \left| \frac{\alpha - \lambda_{\max}}{\alpha + \lambda_{\max}} \right| \right\}.$$

To compute an approximate optimal $\alpha > 0$ such that the convergence factor $\rho(M(\alpha))$ of the HSS-LSPIA iteration is minimized, we can minimize the upper bound $\sigma(\alpha)$ of $\rho(M(\alpha))$ instead. If α^* is such a minimum point, then it must satisfy $\alpha^* - \lambda_{\min} > 0$, $\alpha^* - \lambda_{\max} < 0$, and

$$\frac{\alpha^* - \lambda_{\min}}{\alpha^* + \lambda_{\min}} = \frac{\lambda_{\max} - \alpha^*}{\lambda_{\max} - \alpha^*}.$$

Therefore,

$$\alpha^* = \sqrt{\lambda_{\max}(\mu B^T B)\lambda_{\min}(\mu B^T B)}$$

or

$$\alpha^* = \mu \sqrt{\lambda_{\max}(B^T B)\lambda_{\min}(B^T B)}.$$

\square

Remark 3.6. From (3.16) and Corollary 3.5, we will get $\frac{\mu}{\alpha} = \frac{\mu}{\mu \sqrt{\lambda_{\max}(B^T B) \lambda_{\min}(B^T B)}} = \frac{1}{\sqrt{\lambda_{\max}(B^T B) \lambda_{\min}(B^T B)}}$. Then, we use the $\omega = \frac{1}{\sqrt{\lambda_{\max}(B^T B) \lambda_{\min}(B^T B)}}$ for compute in (3.16) instead of $\frac{\mu}{\alpha}$.

3.3. The surface case

Let two NTP bases $\{B_i(t_{i'})\}_{i,i'=1}^{n_1, m_1} \in \mathbb{C}^{m_1 \times n_1}$, $\{N_j(v_{j'})\}_{j,j'=1}^{n_2, m_2} \in \mathbb{C}^{m_2 \times n_2}$, and a data set points $\{Q_{i',j'}\}_{i',j'=1}^{m_1, m_2} \in \mathbb{R}^{(m_1 \times m_2) \times 3}$ to be fitted, where $m_1 \geq m_2 \geq n_1 \geq n_2$. The initial blending surface $C^0(t, v)$ and $C^{\frac{1}{2}}(t, v)$ can then be generated using all these data points to be the initial control points $\{P_{i,j}^0\}_{i,j=1}^{n_1, n_2} \in \mathbb{R}^{(n_1 \times n_2) \times 3}$ and $\{P_{i,j}^{\frac{1}{2}}\}_{i,j=1}^{n_1, n_2}$, with a real increasing parameters set $\{t_{i'}\}_{i'=1}^m$ i.e. $0 = t_1 < t_2 < t_3 < \dots < t_m = 1$. The initial blending surface can be formed as follows:

$$C^0(t, v) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t) N_j(v) P_{i,j}^0, \quad t, v \in [0, 1], \quad (3.25)$$

and

$$C^{\frac{1}{2}}(t, v) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t) N_j(v) P_{i,j}^{\frac{1}{2}}, \quad t, v \in [0, 1]. \quad (3.26)$$

Let $D = B^T B \otimes N^T N$ where \otimes is Kronecker product. Then, the remaining surfaces of the sequence $C^{k+1}(t, v)$ for $k \geq 0$, can be compute as follows:

$$C^{k+1}(t, v) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t) N_j(v) P_{i,j}^{k+1}, \quad (3.27)$$

where

$$\begin{cases} P_{i,j}^{k+1} &= P_{i,j}^{k+\frac{1}{2}} + \Delta_{i,j}^k \\ \Delta_{i,j}^k &= \frac{\mu}{\alpha} D^T (Q_{i,j} - C^{k+\frac{1}{2}}(t_i, v_j)) \\ P_{i,j}^{k+\frac{1}{2}} &= P_{i,j}^k + \frac{\mu}{\alpha} D^T (Q_{i,j} - C^{k+\frac{1}{2}}(t_i, v_j)). \end{cases} \quad (3.28)$$

We can use the following expression in (3.28) as in the case of curves:

$$C^{k+\frac{1}{2}}(t_i, v_j) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} B_i(t) N_j(v) P_{i,j}^{k+\frac{1}{2}}, \quad k = 0, 1, 2, \dots \quad (3.29)$$

If (3.28) converge, then

$$\lim_{k \rightarrow \infty} P_{i,j}^k = \lim_{k \rightarrow \infty} P_{i,j}^{k+\frac{1}{2}} = \lim_{k \rightarrow \infty} P_{i,j}^{k+1} \quad (3.30)$$

the HSS-LSPIA format of surfaces, including two iterations and substituting the sequential iterative process length from each control point in LSPIA, is recognized as (3.28).

3.4. Convergence analysis of the HSS-LSPIA surface fitting

Lemma 3.7. [1] Given an NTP bases $\{B_i(t_{i'})\}_{i,i'=1}^{n_1, m_1} \in \mathbb{C}^{m_1 \times n_1}$ and $\{N_j(v_{j'})\}_{j,j'=1}^{n_2, m_2} \in \mathbb{C}^{m_2 \times n_2}$, $B^T B$ and $N^T N$ are nonsingular collocation matrices, where $m_1 \geq m_2 \geq n_1 \geq n_2$. Assuming that $\lambda_i(B^T B)$, $i = 1, 2, \dots, n_1$ and $\lambda_i(N^T N)$, $i = 1, 2, \dots, n_2$ are their eigenvalues respectively. Then,

- (1) $0 < \lambda_i(B^T B) \leq 1, 0 < \lambda_i(N^T N) \leq 1$,
- (2) $0 < \lambda_i(D) \leq 1$, $D = B^T B \otimes N^T N$ and here \otimes is Kronecker product.

Let $D = \hat{M}_i - \hat{N}_i$ ($i = 1, 2$) be two splitting of the matrix D . The two iterative processes of (3.28) can be written in the matrix form

$$\begin{cases} \hat{M}_1 P^{k+\frac{1}{2}} = \hat{N}_1 P^k + \mu D^T Q, \\ \hat{M}_2 P^{k+1} = \hat{N}_2 P^{k+\frac{1}{2}} + \mu D^T Q, \end{cases} \quad (3.31)$$

where $\hat{M}_1 = (\alpha I + \mu D^T D)$, $\hat{N}_1 = \alpha I$, $\hat{M}_2 = \alpha I$, $\hat{N}_2 = (\alpha I - \mu D^T D)$, I is identity matrix. Then we can rewrite (3.31) as a unified matrix iteration format as follows

$$P^{k+1} = \hat{M}_2^{-1} \hat{N}_2 \hat{M}_1^{-1} \hat{N}_1 P^k + \mu \hat{M}_2^{-1} (I + \hat{N}_2 \hat{M}_1^{-1}) D^T Q, \quad k = 0, 1, 2, \dots \quad (3.32)$$

or

$$P^{k+1} = \hat{M} P^k + \mu \hat{M}_0 D^T P^0, \quad k = 0, 1, 2, \dots \quad (3.33)$$

where $\hat{M} = \hat{M}_2^{-1} \hat{N}_2 \hat{M}_1^{-1} \hat{N}_1$ and $\hat{M}_0 = \hat{M}_2^{-1} (I + \hat{N}_2 \hat{M}_1^{-1})$.

Now we need to show that the iterative control point sequence $\{P^{k+1}\}$ leads to the unique solution P^* , i.e. $\rho(\hat{M}) < 1$.

Theorem 3.8. Let D be a positive definite matrix, let $\hat{H} = \frac{\mu}{2} (D^T D + (D^T D)) = \mu D^T D$ and $\hat{S} = \frac{\mu}{2} (D^T D - (D^T D)) = 0$ be its Hermitian and skew-Hermitian part, and let α be a positive constant. The HHS-LSPIA iteration's matrix $\hat{M}(\alpha)$ is then provided by

$$\hat{M}(\alpha) = (\alpha I + \hat{S})^{-1} (\alpha I - \hat{H}) (\alpha I + \hat{H})^{-1} (\alpha I - \hat{S})$$

or

$$\hat{M}(\alpha) = (\alpha I)^{-1} (\alpha I - \mu D^T D) (\alpha I + \mu D^T D)^{-1} (\alpha I) \quad (3.34)$$

and the spectral radius of $\hat{M}(\alpha)$ is $\rho(\hat{M}(\alpha))$, bounded by

$$\sigma(\alpha) \equiv \max_{\lambda_i \in \lambda(\hat{H})} \left| \frac{\alpha - \lambda_i}{\alpha + \lambda_i} \right|,$$

where $\lambda(\hat{H})$ is the spectral set of the matrix \hat{H} .

As a consequence, it maintains that

$$\rho(\hat{M}(\alpha)) \leq \sigma(\alpha) < 1 \quad \forall \alpha > 0,$$

i.e., the HHS-LSPIA iteration converges to the unique solution P^* of the linear equation (3.3).

Proof. By the resemblance invariance to the matrix spectrum, we get

$$\begin{aligned} \rho(\hat{M}(\alpha)) &= \rho((\alpha I)^{-1} (\alpha I - \mu B^T B) (\alpha I + \mu B^T B)^{-1} (\alpha I)) \\ &= \rho((\alpha I - \mu D^T D) (\alpha I + \mu D^T D)^{-1} (\alpha I) (\alpha I)^{-1}) \\ &\leq \|(\alpha I - \mu D^T D) (\alpha I + \mu D^T D)^{-1} (\alpha I) (\alpha I)^{-1}\|_2 \\ &\leq \|(\alpha I - \mu D^T D) (\alpha I + \mu D^T D)^{-1}\|_2 \cdot \|(\alpha I) (\alpha I)^{-1}\|_2. \end{aligned}$$

We know that $(\alpha I) (\alpha I)^{-1} = I$ which is unitary matrix i.e. , $\|(\alpha I) (\alpha I)^{-1}\|_2 = 1$. Then , we get

$$\begin{aligned} \rho(\hat{M}(\alpha)) &< \|(\alpha I - \mu D^T D) (\alpha I + \mu D^T D)^{-1}\|_2 \\ &= \max_{\lambda_i \in \lambda(\hat{H})} \left| \frac{\alpha - \lambda_i}{\alpha + \lambda_i} \right|. \end{aligned}$$

Since $\lambda_i > 0$ ($i = 1, 2, \dots, n$) and α is positive constant, it is simpler to see that $\rho(\hat{M}(\alpha)) \leq \rho(\alpha) < 1$. \square

Corollary 3.9. Under the same assumption of Theorem 3.8. Let λ_{max} and λ_{min} be the maximum and the minimum eigenvalues of $D^T D$, respectively. Then, we have the fastest convergence rate of HHS-LSPIA method for the surface fitting when

$$\alpha^* = \mu \sqrt{\lambda_{max}(D^T D)\lambda_{min}(D^T D)} \quad (3.35)$$

where $\mu = \frac{2}{\lambda_{max} + \lambda_{min}}$ presented in [3].

The proof to the Corollary 3.9 is omitted since it is similar to the case of Corollary 3.5.

Remark 3.10. From (3.28) and Corollary 3.9, we will get $\frac{\mu}{\alpha} = \frac{\mu}{\mu \sqrt{\lambda_{max}(D^T D)\lambda_{min}(D^T D)}} = \frac{1}{\sqrt{\lambda_{max}(D^T D)\lambda_{min}(D^T D)}} = \hat{\omega}$. Then, we use the $\hat{\omega} = \frac{1}{\sqrt{\lambda_{max}(D^T D)\lambda_{min}(D^T D)}}$ for compute in (3.28) instead of $\frac{\mu}{\alpha}$.

4. Implementation and examples

In this section, we demonstrated how the HHS-LSPIA effectiveness for three B-spline curve fitting examples and three examples of B-spline tensor product surface fitting as follows:

- Example 1: A subdivision curve generated using the incenter subdivision technique yielded 100 points (music notation).
- Example 2: $y = 3\cos(\theta)$ ($\theta \in [0, 6.28]$) is a sample of 68 points taken evenly from the polar coordinate equation.
- Example 3: G-shape font characteristics measured and smoothed with 315 points.
- Example 4: $z = y^2\cos(x)$ the surface model with 21×21 (441) points.
- Example 5: $z = y\sin(x^2) - x\cos(y^2)$ the surface model with 21×21 (441) points.
- Example 6: $z = x^2\sin(y^2)$ the surface model with 33×33 (1089) points.

For all the examples, the number of control points is equal to the number of data points and the number of control points is the same in each method. We show the details of the implementation for B-spline curve fitting with the HSS-LSPIA method in the following sections, and we deduce the specifics surface for B-spline tensor product surface fitting in just the same way. Given an ordered point set $\{Q_j\}_{j=1}^m$, we assign the parameters $\{t_j\}_{j=1}^m$ for $\{Q_j\}_{j=1}^m$ with the normalized accumulated chord parameterization method [3, 20], that is, $t_1 = 0$, $t_m = 1$,

$$t_j = t_{j-1} + \frac{\|Q_j - Q_{j-1}\|}{D}, \quad j \in \{2, 3, 4, \dots, m-1\}, \quad (4.1)$$

where $D = \sum_{j=2}^m \|Q_j - Q_{j-1}\|$ is the sum of chord length. In addition, the knots for the cubic B-spline fitting curve $C(t) = \sum_{i=1}^n B_{i,3}(t)P_i$, is defined as $\{0, 0, 0, 0, \bar{t}_4, \bar{t}_5, \dots, \bar{t}_n, 1, 1, 1, 1\}$, where

$$\bar{t}_{i+3} = (1 - \alpha)t_{j-1} + \alpha t_j \quad (4.2)$$

where $d = \frac{m}{n-2}$, $\alpha = id - i$, $j = \lfloor id \rfloor$ $i \in \{1, 2, 3, \dots, n-3\}$.

In our implementation, we choose the parameter α^* defined by (3.24) for the HSS-LSPIA method. The iteration process is stopped for the comparison of the LSPIA [3], ELSPIA [6], WHPIA [9] and HSS-LSPIA methods if $\|E_{k+1} - E_k\| < 10^{-7}$, where the fitting error of the k -th iteration is computed by

$$E_k = \sum_{j=1}^m \|Q_j - \sum_{i=1}^n B_i(t_j)P_i^k\|^2, \quad k \geq 0 \quad (4.3)$$

and we compare E_k of each iteration to show the efficiency and validity of the HSS-LSPIA.

All the examples are operated on a laptop with a 2.4 GHz Quad-Core Intel Core i5 processor and 8 GB memory via MATLAB R2019a.

In Examples 1–3, the cubic B-spline curves created by the LSPIA [3], ELSPIA [6], WHPIA [9] and HSS-LSPIA methods with the fitting result are shown in Figures 1–3 respectively. In Examples 4–6, the B-spline tensor product surface fitting created by the LSPIA [3], ELSPIA [6], WHPIA [9] and HSS-LSPIA methods with the fitting result are shown in Figures 4–6 respectively. The blue points in each of the three instances represent known points that form a dotted limit curve to be fitted, green points represent control points gained at the appropriate step, and the red line shows the curve of each iteration.

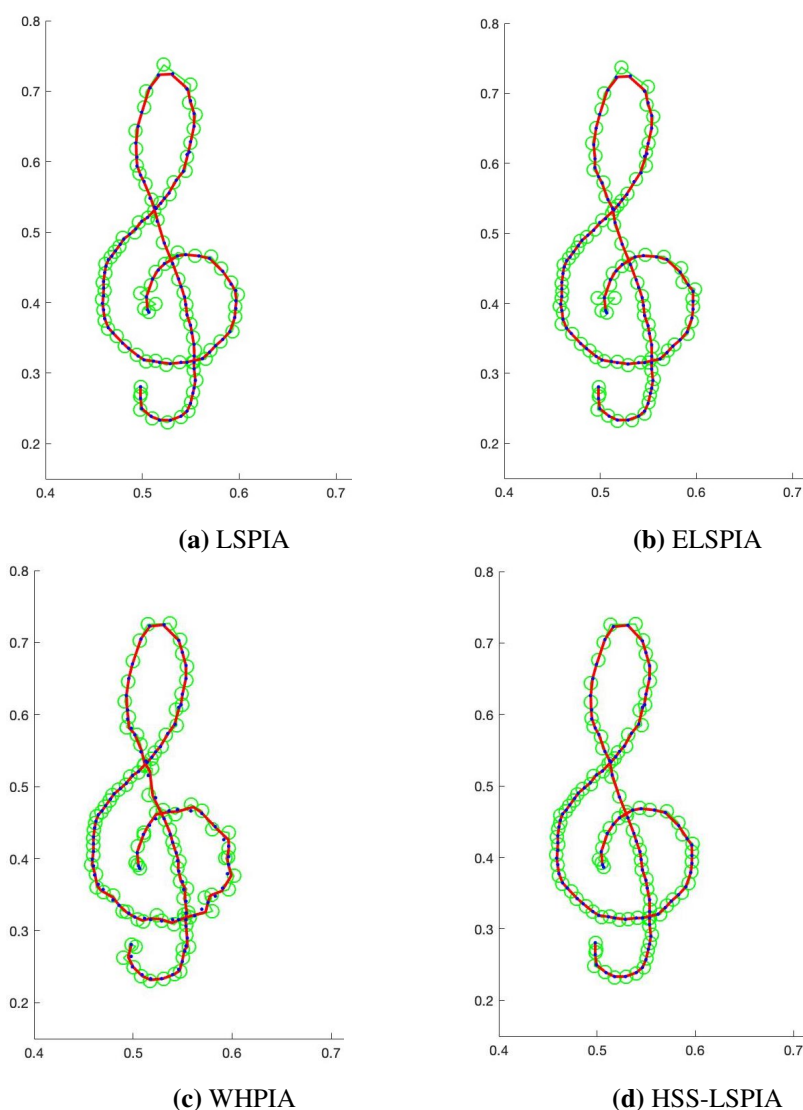


Figure 1. The fitting results of Example 1 using LSPIA, ELSPIA, WHPIA and HSS-LSPIA.

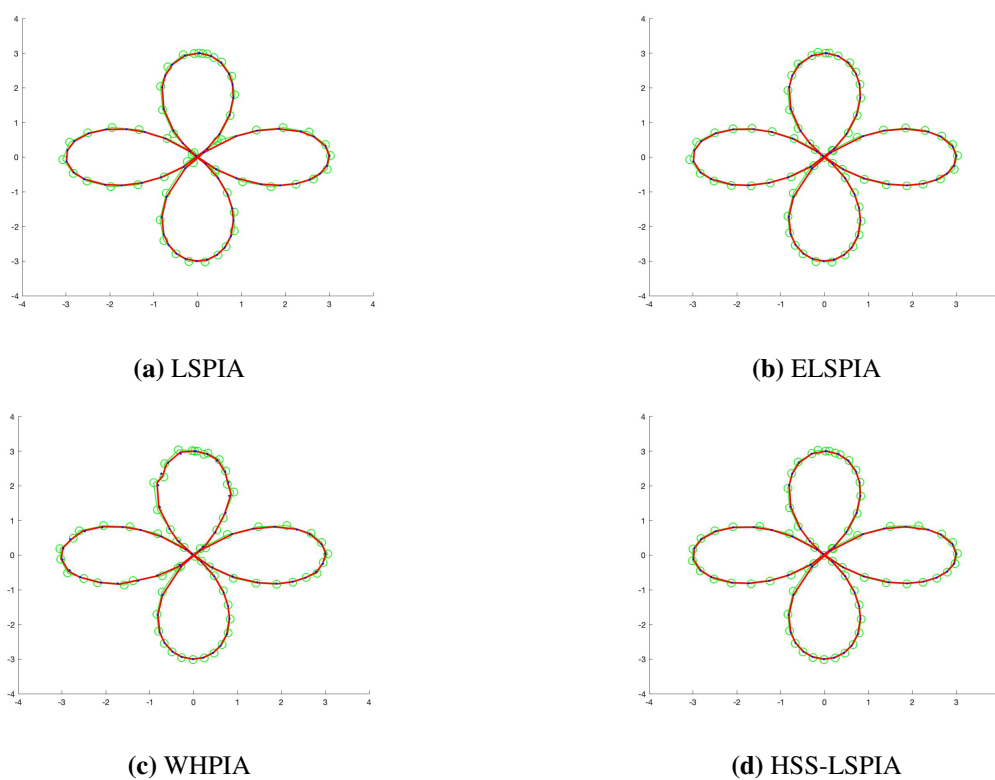


Figure 2. The fitting results of Example 2 using LSPIA, ELSPiA, WHPIA and HSS-LSPIA.

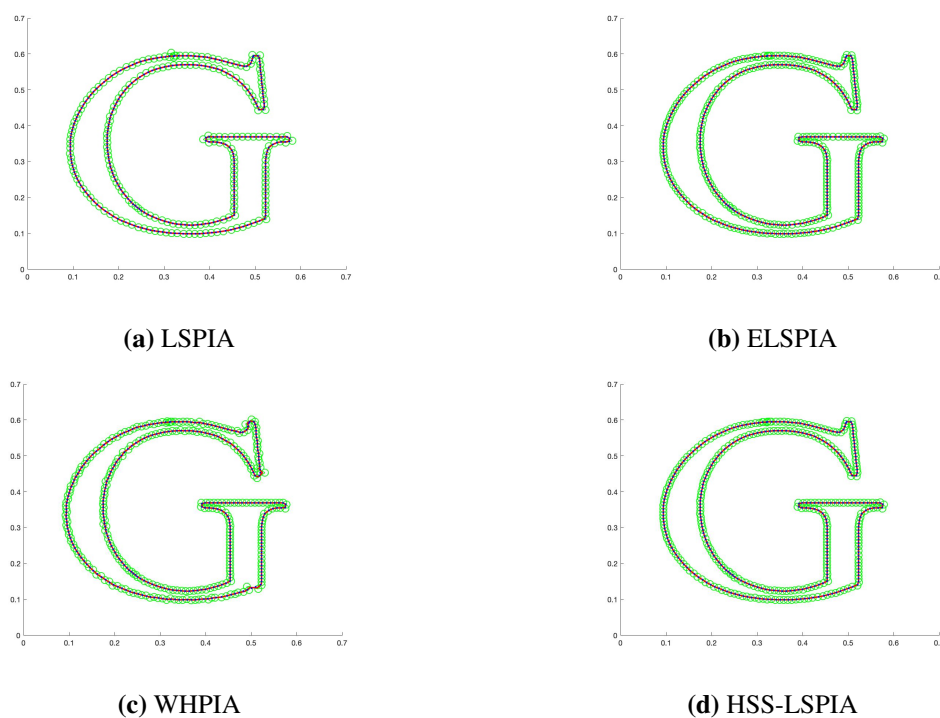
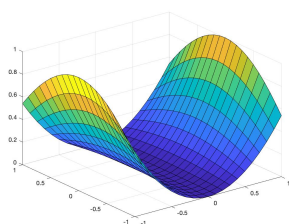
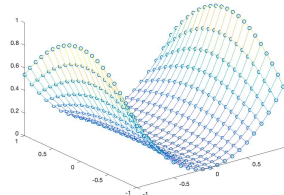


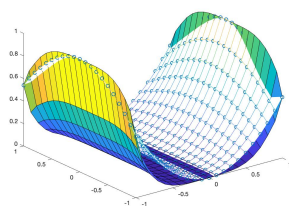
Figure 3. The fitting results of Example 3 using LSPIA, ELSPiA, WHPIA and HSS-LSPIA.



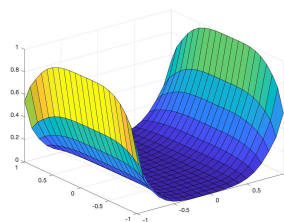
(a) Original surface



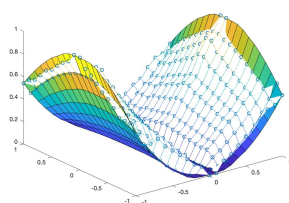
(b) Data points



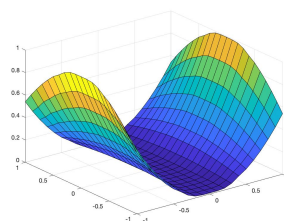
(c) Control points at the fitting result using LSPIA



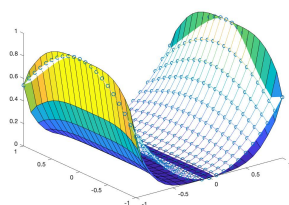
(d) Fitting result using LSPIA



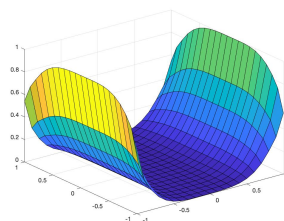
(e) Control points at the fitting result using ELSPIA



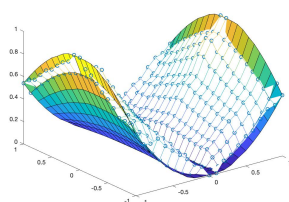
(f) Fitting result using ELSPIA



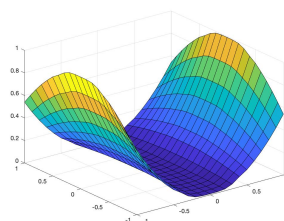
(g) Control points at the fitting result using WHPIA



(h) Fitting result using WHPIA

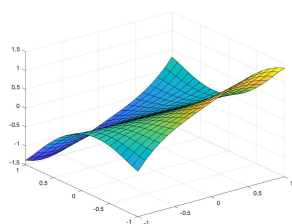


(i) Control points at the fitting result using HSS-LSPIA

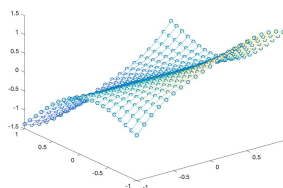


(j) Fitting result using HSS-LSPIA

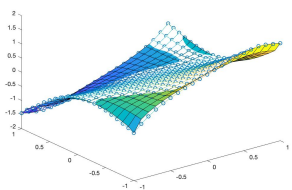
Figure 4. The fitting results of Example 4 using LSPIA, ELSPIA, WHPIA and HSS-LSPIA, where 4c and 4d shows the result of LSPIA, 4e and 4f shows the result of ELSPIA, 4g and 4h shows the result of WHPIA and 4i and 4j shows the result of HSS-LSPIA.



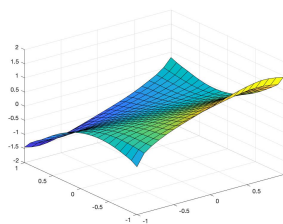
(a) Original surface



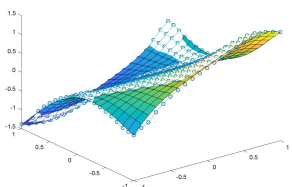
(b) Data points



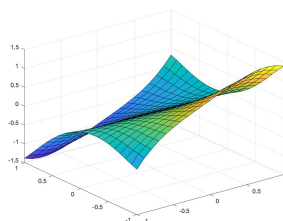
(c) Control points at the fitting result using LSPIA



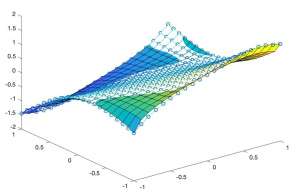
(d) Fitting result using LSPIA



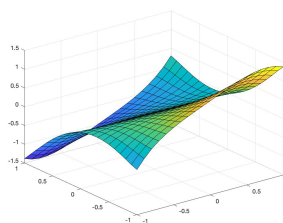
(e) Control points at the fitting result using ELSPIA



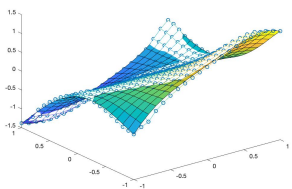
(f) Fitting result using ELSPIA



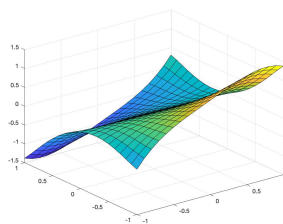
(g) Control points at the fitting result using WHPIA



(h) Fitting result using WHPIA

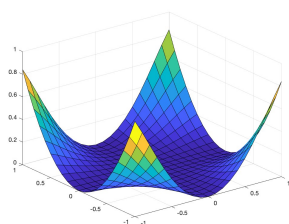


(i) Control points at the fitting result using HSS-LSPIA

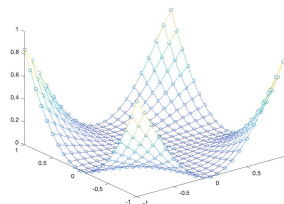


(j) Fitting result using HSS-LSPIA

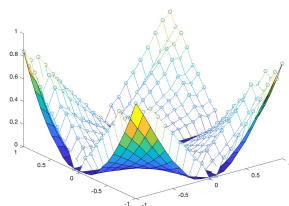
Figure 5. The fitting results of Example 5 using LSPIA, ELSPIA, WHPIA and HSS-LSPIA, where 5c and 5d shows the result of LSPIA, 5e and 5f shows the result of ESPIA, 5g and 5h shows the result of WHPIA and 5i and 5j shows the result of HSS-LSPIA.



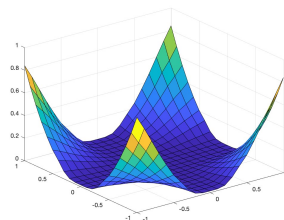
(a) Original surface



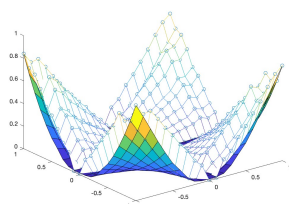
(b) Data points



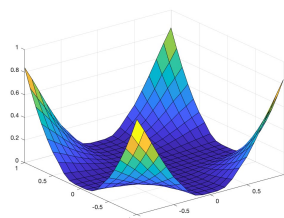
(c) Control points at the fitting result using LSPIA



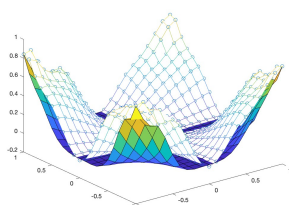
(d) Fitting result using LSPIA



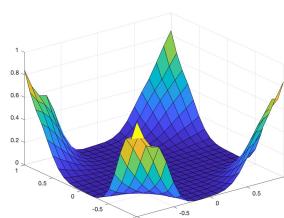
(e) Control points at the fitting result using ELSPIA



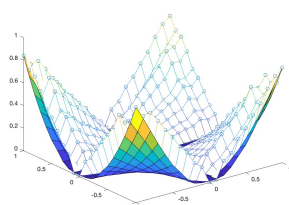
(f) Fitting result using ELSPIA



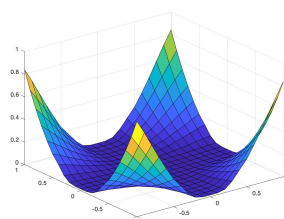
(g) Control points at the fitting result using WHPIA



(h) Fitting result using WHPIA



(i) Control points at the fitting result using HSS-LSPIA



(j) Fitting result using HSS-LSPIA

Figure 6. The fitting results of Example 6 using LSPIA, ELSPIA, WHPIA and HSS-LSPIA, where 6c and 6d shows the result of LSPIA, 6e and 6f shows the result of ELSPIA, 6g and 6h shows the result of WHPIA and 6i and 6j shows the result of HSS-LSPIA.

In Figure 7a–7f, we can see that the HSS-LSPIA method has the fitting inaccuracy of the k -th iteration less than the ELSPIA method, LSPIA method and the WHPIA method respectively.

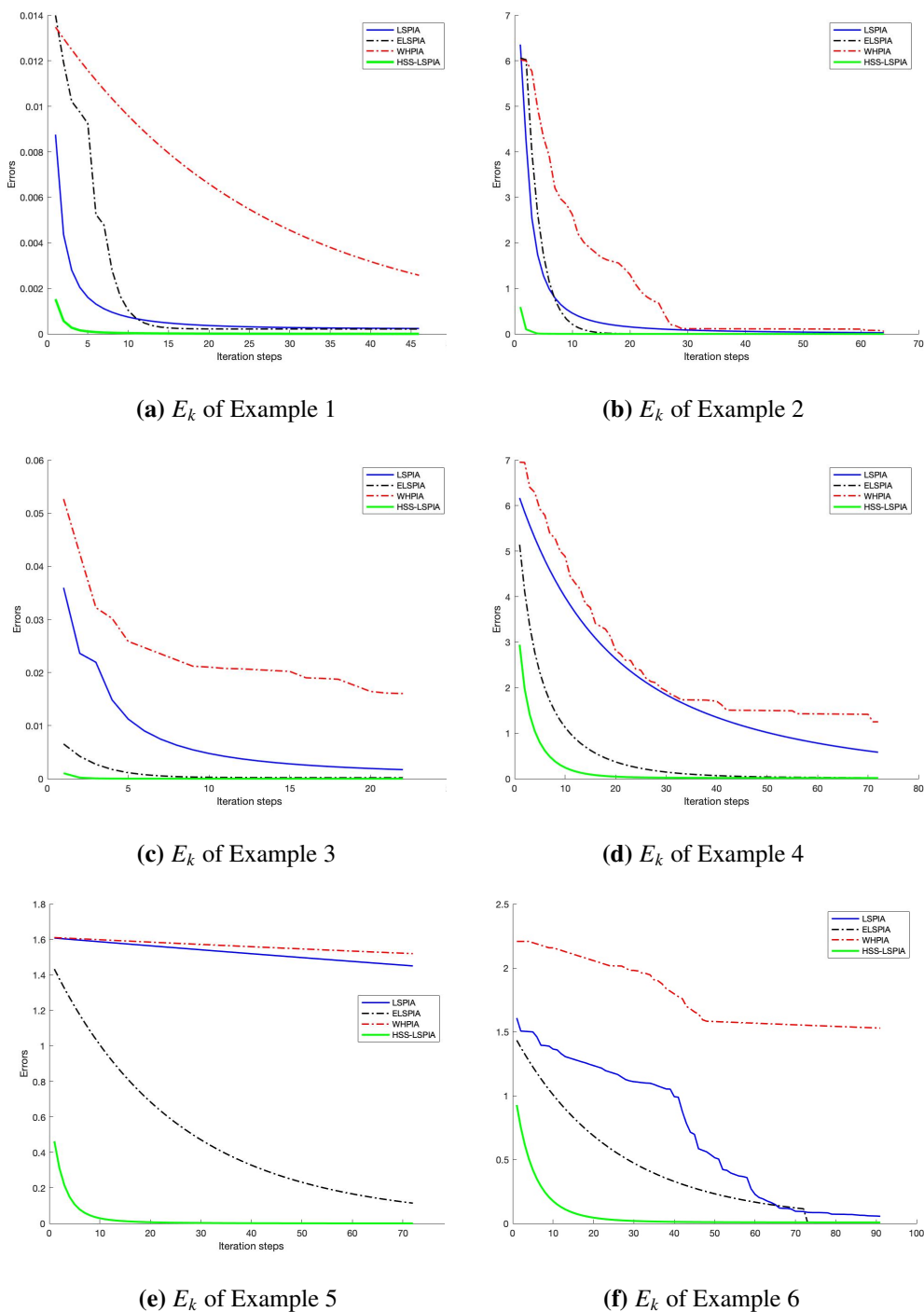


Figure 7. The comparison of errors for Examples 1–6.

In Table 1 columns “IT” (iteration numbers), “CPU time(s)” (ten times the average amount of CPU times(s)) and “ $\|E_{k+1}\|$ ” (fitting error of the $(k+1)$ th iteration) for the LSPIA [3], ELSPIA [6], WHPIA [9] and HSS-LSPIA methods are listed. We can observe that the HSS-LSPIA methods requires fewer

iteration numbers, uses less CPU time, and has a smallest fitting error at $(k + 1)$ th iteration. We can now conclude from the experiments that, the HSS-LSPIA methods is faster than the ELSPIA method, LSPIA method and the WHPIA method respectively, in terms of convergence speed. The examples provided in this section are just intended to show how effective the methods propose by us. Because the spectral radius will vary depends on the NTP bases, we cannot guarantee that our methods will converge faster than ELSPIA, LSPIA, and WHPIA in other examples.

Table 1. Comparison of IT, CPU time(s) and $\|E_{k+1}\|$.

Examples	Methods	IT	CPU time(s)	$\ E_{k+1}\ $
Example 1	LSPIA	101	0.0084870	1.6400E-05
	ELSPIA	59	0.0037290	1.5100E-05
	WHPIA	130	0.0082561	1.9899E-05
	HSS-LSPIA	47	0.0020687	3.0800E-06
Example 2	LSPIA	410	0.0069510	3.2800E-05
	ELSPIA	105	0.0044940	4.7500E-05
	WHPIA	1877	0.0319717	5.2800E-04
	HSS-LSPIA	65	0.0013589	3.7900E-06
Example 3	LSPIA	395	0.0458660	1.6100E-05
	ELSPIA	85	0.0091790	1.3100E-05
	WHPIA	763	0.0747946	2.4400E-04
	HSS-LSPIA	23	0.0059934	1.1220E-06
Example 4	LSPIA	470	5.5093410	2.7500E-04
	ELSPIA	137	1.7922920	1.6703E-04
	WHPIA	2901	64.869787	2.0210E-03
	HSS-LSPIA	73	1.2409530	1.6634E-05
Example 5	LSPIA	477	6.887048	3.9281E-04
	ELSPIA	407	5.638451	2.7947E-04
	WHPIA	3834	89.834053	1.9947E-03
	HSS-LSPIA	74	1.335767	2.5180E-05
Example 6	LSPIA	1220	42.263918	2.0272E-04
	ELSPIA	623	7.318428	1.0700E-04
	WHPIA	3935	68.133107	5.6847E-03
	HSS-LSPIA	96	1.596564	3.7280E-05

5. Conclusions

In this research, based on the HSS iterative approach for solving system of linear equations of LSPIA, a new progressive and iterative approximation for least square fitting method called HSS-LSPIA is developed to solve the problems of curves and surfaces approximation with NTP bases. The HSS-LSPIA format including two iterations with the iterative distinction vectors in which the two iterations are different from each other. And then, we provided the approximate value of the fastest convergence positive constant. In terms of iterative steps and computing time, numerical

results indicate that the HSS-LSPIA method is more effectively efficient than the ELSPIA, LSPIA, and WHPIA methods, respectively. It has demonstrated that the HSS-LSPIA method can be faster than the ELSPIA, LSPIA, and WHPIA methods in terms of convergence speed.

Acknowledgments

The authors acknowledge the financial support provided by the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT. The first author was supported by the Petchra Pra Jom Klao Ph.D. Research Scholarship from King Mongkut's University of Technology Thonburi (Grant No. 54/2561). Moreover, this research project is supported by Thailand Science Research and Innovation (TSRI) Basic Research Fund: Fiscal year 2022 under project number FRB650048/0164.

Conflict of interest

The authors declare that there is no conflict of interest regarding the publication of this article.

References

1. H. W. Lin, H. J. Bao, G. J. Wang, Totally positive bases and progressive iteration approximation, *Comput. Math. Appl.*, **50** (2005), 575–586. <https://doi.org/10.1016/j.camwa.2005.01.023>
2. H. Lin, Z. Zhang, An extended iterative format for the progressive-iteration approximation, *Comput. Graph.*, **35** (2011), 967–975. <https://doi.org/10.1016/j.cag.2011.07.003>
3. C. Deng, H. Lin, Progressive and iterative approximation for least squares B-spline curve and surface fitting, *Comput.-Aided Des.*, **47** (2014), 32–44. <https://doi.org/10.1016/j.cad.2013.08.012>
4. A. Ebrahimi, G. B. Loghmani, A composite iterative procedure with fast convergence rate for the progressive-iteration approximation of curves, *J. Comput. Appl. Math.*, **359** (2019), 1–15. <https://doi.org/10.1016/j.cam.2019.03.025>
5. H. Lin, T. Maekawa, C. Deng, Survey on geometric iterative methods and their applications, *Comput.-Aided Des.*, **95** (2018), 40–51. <https://doi.org/10.1016/j.cad.2017.10.002>
6. H. Wang, On extended progressive and iterative approximation for least squares fitting, *Vis. Comput.*, **38** (2022), 591–602. <https://doi.org/10.1007/s00371-020-02036-8>
7. J. M. Peña, *Shape preserving representations in computer-aided geometric design*, Commack, New York: Nova Science Publishers, 1999.
8. Z. Z. Bai, G. Golub, M. Ng, Hermitian and skew-hermitian splitting methods for non-hermitian positive definite linear systems, *SIAM J. Matrix Anal. Appl.*, **24** (2003), 603–626. <https://doi.org/10.1016/j.laa.2007.02.018>
9. L. Hu, H. Shou, Z. Dai, Hss-iteration-based iterative interpolation of curves and surfaces with NTP bases, In: *Simulation tools and techniques*, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Springer, 2019. https://doi.org/10.1007/978-3-030-32216-8_36

10. H. Lin, Z. Zhang, An efficient method for fitting large data sets using T-splines, *SIAM J. Sci. Comput.*, **35** (2013), A3052–A3068. <https://doi.org/10.1137/120888569>
11. G. H. Golub, C. F. Van Loan, *Matrix computations*, 3 Eds., Baltimore: Johns Hopkins University Press, 1996.
12. Y. Saad, *Iterative methods for sparse linear systems*, 2 Eds., SIAM, Philadelphia, 2003.
13. D. M. Young, *Iterative solution of large linear systems*, New York: Academic Press, 1971.
14. C. Liu, J. Li, L. Hu, Jacobi–PIA algorithm for bi-cubic B-spline interpolation surfaces, *Graph. Models*, **24** (2022), 101134. <https://doi.org/10.1016/j.gmod.2022.101134>
15. F. H. Yusuf, Y. Jiang, H. Lin, Gauss-seidel progressive and iterative approximation for least squares fitting, *J. Comput.-Aided Des. Comput. Graph.*, **33** (2021), 1–10. <https://doi.org/10.3724/SP.J.1089.2021.18289>
16. C. Liu, X. Han, L. Zhang, Unconditional convergence of iterative approximation methods, *Eng. Anal. Bound. Elem.*, **126** (2021), 161–168. <https://doi.org/10.1016/j.enganabound.2021.03.001>
17. G. H. Golub, D. Vanderstraeten, On the preconditioning of matrices with a dominant skew-symmetric component, *Numer. Algorithms*, **25** (2000), 223–239.
18. C. L. Wang, Z. Z. Bai, Sufficient conditions for the convergent splittings of non-hermitian positive definite matrices, *Linear Algebra Appl.*, **330** (2001), 215–218. [https://doi.org/10.1016/S0024-3795\(01\)00275-0](https://doi.org/10.1016/S0024-3795(01)00275-0)
19. Z. Z. Bai, J. F. Yin, Y. F. Su, Shift-splitting preconditioner for non-Hermitian positive definite matrices, *J. Comput. Math.*, **24** (2006), 539–552.
20. I. J. Schoenberg, A. Whitney, On polya frequency function. III. The positivity of translation determinants with an application to the interpolation problem by spline curves, *Trans. Amer. Math. Soc.*, **74** (1953), 246–259.



©2022 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)