Applied Mathematics & Information Sciences
An International Journal

An Extension of a New Kind of Graphics Fitting Method

Shaohui Deng[1], Sujie Guan[2], Guozhao Wang[1] and Shaobo Deng[3,4]

[1] Department of Mathematics, Zhejiang University, Zhejiang 310027, China
[2] School of Information Engineering, NanChang Institute of Technology, Nanchang, 330099 P.R. China
[3]Key Laboratory of intelligent information, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190.P.R. China
[4]Graduate University of Chinese Academy of Sciences, Beijing, 100180, P.R. China

Received: 25 Aug. 2012; Revised 21 Nov. 2012; Accepted 3 Dec. 2012
Published online: 1 Mar. 2013

Abstract: The progressive iterative approximation method (PIA) for fitting is developed in recent years. The previous research is mainly restricted to the normalized totally positive (NTP) basis. In this paper, we propose a new extended method which can unify the classic PIA method, weighted PIA method (WPIA) and local PIA (LPIA) method into one formula by introducing the transform matrix. In addition, the method presented by us has the following advantages: firstly, it can be applied to any basis; secondly, compared with other extended methods, the convergence rate of our method can be independent of the initial parameterizations; thirdly, for the NTP basis, the extended PIA method can be accelerated greatly with the fastest convergence rate. Finally, we present two numerical examples to show that our algorithm is efficient.

Keywords: Progressive-iterative, Data fitting, Transform matrix

1. Introduction

Data fitting is used widely in the field of CAGD and CG. From the history point of view, this topic has been extensively studied. Many kinds of methods are well established, for example, the least-square approximation, the Newton interpolation and other numerical methods. However, the potential disadvantage of the traditional fitting method is that it lacks of geometric intuition, which is important to the CAGD. Recently, a new kind of data fitting method, called the progressive iterative approximation, has received much attention which can compensate for that problem, furthermore it has the virtue of simplicity, no need to solve a system of linear equations and numerical stability and so on, hence, it is more attractive when comparing with the traditional methods.

The PIA method was proposed by Qi et al. [1] and de-Boor [2] respectively. Afterwards, Lin et al. [3,4] showed that the PIA method can be extended to the non-uniform B-spline basis functions and surface, furthermore, to any normalized totally positive basis functions. Likely, Maekawa et al. [5] made an improvement to the PIA method, which can be viewed as the geometric form of the PIA method. Some new progress has been made, for instance, Chen et al. [6] extended this method to triangular Bézier surfaces. Very recently, Lin [7] developed an adaptive algorithm, which is more flexible in data fitting.

So far, three kinds of iterative formats of the PIA method have been presented: Qi [1] presented the classical PIA iterative format; Lin [8] devised a local PIA iterative format which can fit the data points locally; Lu [9] presented weighted iterative format to speed up the convergence of the method. Although there is significant difference between the three kinds of PIA methods in view of the iterative form, we find that they have the unified iterative format by introducing the transform matrix. In addition, we present another iterative format which can be applied to any basis functions including the NTP basis. Especially, as for the NTP basis, the convergence rate of the extended method given by us can be accelerated greatly.

* Corresponding author: e-mail: paperdeng@126.com

2. Relationship between the PIA method and power series expansion of the collocation matrix

2.1. Preliminary

Given a sequence of data points $\{P_i\}_{i=0}^n$ and a set of normalized totally positive basis functions $\{B_i(t)|i=0,1,...,n\}$, each P_i is assigned a parameter value t_i, $i=0,1,\cdots n$, where $\{t_i\}_{i=0}^n$ is a real increasing sequence, that is, $t_0 < t_1 < \cdots < t_n$. For $k=0,1,\cdots$, the iteration of the PIA method consists of the following three steps:

Step1. Generate a starting curve as:

$$R^0(t) = B(t)P^0$$

where $P^0 = [P_0^0, P_1^0, \cdots, P_n^0]^T$ and $P_i^0 = P_i$, $i=0,1,\cdots,n$.

Step2. Compute the adjusting vectors:

$$\Delta_i^k = P_i - R^k(t_i), i=0,1,\cdots,n.$$

Step3. Adjust the control points:

$$P_i^{k+1} = P_i^k + \Delta_i^k, i=0,1,\cdots,n.$$

Repeating this process, we get a curve sequence

$$R^k(t) = B(t)P^k, k=0,1,\cdots.$$

where $P^k = [P_0^k, P_1^k, \cdots, P_n^k]^T$.

We have the following iterative formula:

$$\Delta^k = (I-B)\Delta^{k-1}, k=1,2,\cdots \quad (1)$$

where $\Delta^k = [\Delta_0^k, \Delta_1^k, \cdots, \Delta_n^k]^T$.

2.2. Power series expansion of the inverse of the collocation matrix

In this section, we reveal the relationship between the PIA method and the power series expansion of the matrix inverse. We firstly introduce the following theorem.

Theorem 2.1. Let B and U be the square nonsingular matrices and suppose their spectral radius satisfies: $\rho(I-UB) < 1$, then we have the following power series expansion:

$$B^{-1} = U + \sum_{k=1}^{\infty} (I-UB)^k U, \quad (2)$$

where I is the identity matrix.

Proof. Since matrices B and U satisfy: $\rho(I-UB) < 1$, it is not difficult to see that the matrix $I-UB$ is nonsingular.

Let's consider the following matrix equation:

$$\sum_{i=0}^{s} (I-UB)^i (UB) = I - (I-UB)^{s+1}.$$

In terms of $\rho(I-UB) < 1$, taking the limit, we have

$$\lim_{s\to\infty} (I-UB)^{s+1} = 0,$$

Furthermore, it gives

$$\sum_{i=0}^{\infty} (I-UB)^i (UB) = I.$$

Hence, we can obtain the conclusion:

$$B^{-1} = U + \sum_{k=1}^{\infty} (I-UB)^k U.$$

It should be noted that there always exists such matrix U for any given nonsingular matrix B that the series of the matrix converges. So, we present the following lemma which guarantees the existence of the matrix U.

lemma 2.1. For any given nonsingular matrix, the matrix U always exists. In general, we can take

$$U = \frac{B^T}{\rho(BB^T)},$$

where $\rho(BB^T)$ is the spectral radius of the matrix BB^T.

Theorem 2.1 gives another way to find the matrix inverse, the main challenge of this method is that the convergence rate is uncertain. In general, different choice of the matrix U leads to different convergence rate of the power series expansion of the matrix B^{-1} which depends on the spectral radius $\rho(I-UB) < 1$. In addition, the choice of the matrix U is not unique. For some special form of matrix B, we can choose some special matrix U.

Corollary 2.1. Let B be a nonsingular square matrix and suppose the eigenvalues of the matrix B satisfy: $0 < \lambda_i(B) < 1, i=0,1,\cdots,n$. If we take $U=I$, then we have the following matrix expansion:

$$B^{-1} = \sum_{k=0}^{\infty} (I-B)^k.$$

Corollary 2.2. Let B be a nonsingular square matrix and suppose the eigenvalues of the matrix B satisfy: $0 < \lambda_i(B) < 1, i=0,1,\cdots,n$. If we take $U = aI$ with $0 < a < \frac{1}{\lambda_{\max}(B)}$, then we have the following matrix expansion:

$$B^{-1} = aI + a\sum_{i=1}^{\infty} (I-aB)^i,$$

where $\lambda_{\max}(B)$ is the maximal eigenvalue of the matrix B.

It is obvious that different value a leads to different convergence rate of the matrix expansion. Therefore, we can optimize the value a to speed up convergence of the matrix series. To this end, we introduce the following corollary.

Corollary 2.3. Let B a nonsingular square matrix and suppose the eigenvalues of the matrix B satisfy: $0 < \lambda_i(B) <$

$1, i = 0, 1, \cdots, n$. If we take $a = \frac{2}{1+\lambda_{\min}(B)}$, then it will speed up the convergence of the matrix series. Furthermore, we have the following matrix expansion:

$$B^{-1} = \frac{2}{1+\lambda_{\min}(B)} I + \frac{2}{1+\lambda_{\min}(B)} \sum_{i=1}^{\infty} \left(I - \frac{2}{1+\lambda_{\min}(B)} B \right)^i.$$

where $\lambda_{\min}(B)$ is the minimal eigenvalue of the matrix B.

Proof. The proof is similar to the [9], here we leave details to the interested readers.

From the three corollaries above, we find that if different form of the matrix U is taken, the different power series expansion of the matrix inverse can be obtained. If we take some special form of the matrix U, some special matrix expansion can be obtained.

Corollary 2.4. Let B be a nonsingular square matrix. Suppose the matrix U satisfies:

$$UB = \begin{bmatrix} I & 0 \\ C_{21} & C_{22} \end{bmatrix}, \tag{3}$$

and its spectral radius satisfies:$\rho(I - UB) < 1$, then we have

$$B^{-1} = \begin{bmatrix} I \\ C_{21} \; C_{22} \end{bmatrix} + \sum_{i=1}^{\infty} (I - UB)^i U.$$

2.3. Interpolation expressed by the matrix expansion

The interpolation problem is usually stated as: given a sequence of data points $\{P_i\}_{i=0}^n$ and a set of basis functions $\{B_i(t)\}_{i=0}^n$, each P_i is associated with a parameter value t_i, with $0 \leqslant i \leqslant n$, where no two t_i are the same. Generally, the interpolation problem can be expressed in the matrix form

$$\begin{bmatrix} B_0(t_0) & B_1(t_0) & \cdots & B_n(t_0) \\ B_0(t_1) & B_1(t_1) & \cdots & B_n(t_1) \\ \vdots & \vdots & \ddots & \cdots \\ B_0(t_n) & B_1(t_n) & \cdots & B_n(t_n) \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_n \end{bmatrix}.$$

We can shorten the above equation to

$$BX = P,$$

where B is the collocation matrix and $P = [P_0, P_1, \cdots, P_n]^T$.

It is well known that when the matrix B is nonsingular the interpolating curve can be expressed as the following form:

$$P(t) = B(t)B^{-1}P.$$

The interpolation problem can always be reduced to solve a linear system equations. Many numerical methods have been established to solve it. However, we can obtain a special form of the interpolating curve by introducing the transform matrix which is vital to PIA method.

Definition 2.1. Let U be an nonsingular square matrix, $\{B_i(t)\}_{i=0}^n$ be a set of basis functions and $\{P_i\}_{i=0}^n$ be a set of data points. The matrix U is called the transform matrix if we make a transform to the given basis functions and the given data points, that is,

$$\begin{aligned} \overline{B}_j(t) &= \sum_{j=0}^{n} u_{ij} B_j(t) \\ \overline{P}_i &= \sum_{j=0}^{n} u_{ij} P_j \end{aligned} , \tag{4}$$

where u_{ij} is the $ij-$th entry of the matrix U.

According to Theorem 2.1, we can choose the transform matrix U to satisfy $\rho(I - UB) < 1$, then the particular form of the interpolating curve can be obtained.

Theorem 2.2. Given a set of data points $\{P_i\}_{i=0}^n$ and a set of basis functions $\{B_i(t)\}_{i=0}^n$, each P_i is associated with a parameter value with $0 \leqslant t_0 < t_1 < \cdots < t_n \leqslant 1$, suppose U is the transform matrix which satisfies $\rho(I - UB) < 1$, then the interpolating curve can be expressed in the form of matrix expansion. Furthermore, the interpolating curve is independent of the choice of the transform matrix U.

$$R(t) = B(t) \sum_{i=0}^{\infty} (I - UB)^i UP, \tag{5}$$

where B is the collocation matrix.

Proof. Since the matrix U satisfies: $\rho(I - UB) < 1$, it follows that power series of the matrix converges $\sum\limits_{i=0}^{\infty} (I - UB)^i$.

In terms of Theorem 2.1, it is not difficult to see that:

$$\begin{aligned} R(t) &= B(t) \sum_{i=0}^{\infty} (I - UB)^i UP \\ &= B(t)B^{-1}P. \end{aligned}$$

Furthermore, the above conclusion shows that the interpolating curve is independent of the choice of the transform matrix U.

Remark: It should be noted that the interpolating curve $R(t) = B(t)\sum_{i=0}^{\infty}(I - UB)^i UP$ is the same one as that obtained by solving the system of the linear equations, here, we express it in another form. Furthermore, we find that using the Eq.(5), we can derive the three kinds of PIA methods.

Let us rewrite the Eq.(5) as

$$R(t) = B(t)UP + B(t) \sum_{i=1}^{\infty} (I - UB)^i UP.$$

We call the UP the initial control points, and the formula $\sum_{i=1}^{\infty}(I - UB)^i UP$ the iterative control points. The following theorem shows the inner relationship between the PIA method and the matrix expansion.

Theorem 2.3. Given a set of data points and a set of NTP basis of functions $\{B_i(t)\}_{i=0}^n$, the transform matrix U satisfies $\rho(I-UB)<1$, if we set the initial control points $UP=P$, then the PIA method is equivalent to the traditional interpolation method, furthermore we have:

(a) if we take $U=I$, then we can get the classical PIA iterative format.

(b) if we take $U=\frac{2}{1+\lambda_{\min}(B)}I$, then we can get the WPIA iterative format.

(c) if the matrix U is taken as Eq.(3), then we can get the LPIA iterative format.

Proof. Since the initial points are set to P, the interpolating curve can be expressed as the following form:

$$R(t)=B(t)P+B(t)\sum_{i=1}^{\infty}(I-UB)^iUP. \tag{6}$$

We denote $\overline{B}=UB$, $\overline{P}=UP$, then Eq.(6) can be rewritten as

$$R(t)=B(t)P+B(t)\sum_{i=1}^{\infty}(I-\overline{B})^i\overline{P}.$$

Taking $m+1$ and $m+2$ first terms of the above equation, we get

$$R_m(t)=B(t)P+B(t)\sum_{i=1}^{m}(I-\overline{B})^i\overline{P},$$

and

$$R_{m+1}(t)=B(t)P+B(t)\sum_{i=1}^{m+1}(I-\overline{B})^i\overline{P}.$$

Suppose the control points of the approximating curve $R_m(t)$ is Q_i^m , $R_m(t)$ can be rewritten as

$$R_m(t)=\sum_{i=0}^{n}B_i(t)Q_i^m. \tag{7}$$

Suppose the control points of the $R_{m+1}(t)$ is Q_i^{m+1}, by the same way, we can get

$$R_{m+1}(t)=\sum_{i=0}^{n}B_i(t)Q_i^{m+1}. \tag{8}$$

Then, we subtract Eq. (8) from Eq. (7), we can obtain

$$\begin{aligned}R_{m+1}(t)-R_m(t)&=\sum_{i=0}^{n}B_i(t)(Q_i^{m+1}-Q_i^m)\\&=B(t)(I-\overline{B})^{m+1}\overline{P}.\end{aligned}$$

Furthermore, we denote $\overline{\Delta}_i^k, i=0,1,\cdots,n$ by the adjusting errors, like the iterative formula Eq. (1), we can have

$$B(t)(I-\overline{B})^{m+1}\overline{P}=\sum_{i=0}^{n}B_i(t)\overline{\Delta}_i^m.$$

Thus, we get

$$\sum_{i=0}^{n}B_i(t)(Q_i^{m+1}-Q_i^m)=\sum_{i=0}^{n}B_i(t)\overline{\Delta}_i^m.$$

According to $\{B_i(t)\}_{i=0}^n$ are linearly independent, we get:

$$Q_i^{m+1}-Q_i^m=\overline{\Delta}_i^m, i=0,1,\cdots,n, m=0,1,\cdots$$

Since $\overline{\Delta}^k=(I-UB)\overline{\Delta}^{k-1}$, it follows that if we take $U=I$, we obtain the classical PIA method, Similarly, if U is taken to the corresponding forms, we can get cases (b) and (c), which completes the proof.

3. Extended PIA method

Theorem 2.3 shows that the PIA method in essence is equivalent to the following iterative process:

$$\sum_{i=0}^{n}(I-UB)^iU\to B^{-1},$$

where U is the transform matrix. If U is properly chosen, some particular effect can be obtained, for instance, the LPIA and the WPIA method. In this section, we extend the PIA method to any basis functions. The primary challenge in extending the PIA method is the selection of the transform matrix U. To get the most benefit from the choice of this matrix, we would like to find optimal U, minimizing the spectral radius $\rho(I-B)$. The following lemma can be found in the book [10].

Lemma 3.1. Let A be an $n\times n$ matrix, then the matrix A can be diagonalized as:

$$A=QR,$$

where $Q^TQ=I$ and R is the upper triangular matrix.

Let B be the collocation matrix, according to Lemma 3.1, the matrix B can be diagonalized as:

$$\begin{aligned}B&=Q\begin{bmatrix}\lambda_0 & r_{12} & \cdots & r_{1n}\\ & \lambda_1 & \cdots & r_{2n}\\ & & \ddots & \vdots\\ & & & \lambda_n\end{bmatrix}\\&=QR.\end{aligned}$$

We now define the transform matrix U to be:

$$\begin{aligned}U&=\begin{bmatrix}1/\lambda_0 & & & \\ & 1/\lambda_1 & & \\ & & \ddots & \\ & & & 1/\lambda_n\end{bmatrix}Q^T\\&=HQ^T.\end{aligned} \tag{9}$$

The extended PIA method consists of the following steps:

Step1. Input the data points

$$P_i, i = 0, 1, \cdots, n.$$

Step2. Make a transform to the given basis functions and data points.

$$\overline{B}(t) = UB(t)^T$$
$$\overline{P} = UP.$$

Step3. Generate a starting curve as:

$$\overline{R}(t) = \overline{B}(t)\overline{P}.$$

Step4. Compute the error vectors:

$$\overline{\Delta}_i^k = \overline{P}_i - \overline{R}(t_i), i = 0, 1, \cdots, n.$$

Step5. Adjust the control points:

$$\overline{P}_i^{k+1} = \overline{P}_i^k + \overline{\Delta}_i^k, i = 0, 1, \cdots, n.$$

Then, we get the iterative curve sequence:

$$\left\{\overline{R}_n(t)\right\}.$$

Theorem 3.1. Given a set of basis functions $\{B_i(t)\}_{i=0}^n$ and a set of data points $\{P\}_{i=0}^n$, suppose the transform matrix is chosen as Eq. (9), then the extended PIA method converges and this algorithm terminates at most $n+1$ steps.

Proof. According to the algorithm given by us, the iterative curve sequence $\left\{\overline{R}_n(t)\right\}$ can be written as the following form:

$$\overline{R}_n(t) = B(t)\sum_{i=0}^{n}(I - \overline{B})^i\overline{P},$$

where $\overline{B} = UB$ and $\overline{P} = UP$.

Since the matrix B can be diagonalized as $B = QR$ with $QQ^T = I$, it follows that

$$I - \overline{B} = I - HQ^TQR$$
$$= I - HR.$$

It is not difficult to see that the main diagonal of the matrix $I - HR$ is equal to zero, furthermore we have:

$$\rho(I - \overline{B}) = 0.$$

Since $\overline{B}$ is an $(n+1 \times n+1)$ square matrix, it follows that:

$$\left(I - \overline{B}\right)^{n+1} = 0.$$

Then, the error vectors

$$\overline{\Delta}_i^{n+1} = 0, i = 0, 1, \cdots, n.$$

So, the extended PIA method can terminate at most $n+1$ steps.

4. Numerical examples

In this section, we apply our method to two different kinds of basis functions. In order to compare the convergence rate with the previous research, we use the same examples proposed in Delgado [11] and Chen [12]. For the example 1, we use the Bernstein basis $\{B_i^{10}(t)\}_{i=0}^{10}$ which is the typical NTP basis. For example 2, we consider the Wang-Ball basis $\{W_i^4(t)\}_{i=0}^4$ which is non-NTP basis when $n = 4$.

Example 1. Consider the planar Bézier curve of degree 10, its control points are sampled from the Lemniscate of Gerono given by the parametric form

$$(x(t), y(t)) = (\cos(t), \sin(t)\cos(t)), \quad t \in [0, 2\pi]$$

which are sampled by the following way

$$P_i = (x(u_i), y(u_i), u_i = -\frac{\pi}{2} + \frac{i}{10}, i = 0, 1, \cdots 10.$$

Example 2. Consider four degree Wang-ball basis functions, which is defined as follow:

$$(W_0^4, W_1^4, \cdots, W_4^4) = ((1-t)^2, 2(1-t)^3, 4(1-t)^2t^2, t^2)$$

its control points are sampled from the following parametric curve by the the way

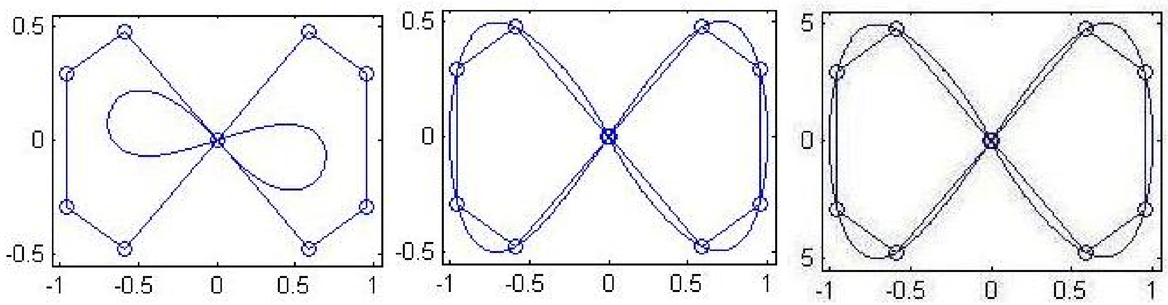
$$P_i = (\cos(t_i), \sin(t_i)), \quad t_i = \frac{i\pi}{4}, i = 0, 1, \cdots, 4.$$

For simplicity, we use the uniform parameter. Each P_i is assigned a parameter value t_i, $t_i = i/n$, $i = 0, 1, \cdots, n$, that is, the uniform distribution of the parameters. Let's consider example 1. Through computation, the transform matrix U is:

$$\begin{bmatrix}
0.88 & 0.31 & 0.09 & 0.02 & 0.01 & 0.0 & 0.00 & 0.00 & 0.00 & 0.00 & 0 \\
-0.69 & 1.57 & 1.18 & 0.55 & 0.18 & 0.05 & 0.01 & 0.00 & 0.00 & 0.00 & 0 \\
0.53 & -2.32 & 1.79 & 2.81 & 1.83 & 0.75 & 0.19 & 0.03 & 0.00 & 0.00 & 0 \\
-0.39 & 2.38 & -4.55 & 0.47 & 4.85 & 4.43 & 2.09 & 0.52 & 0.05 & 0.01 & 0 \\
0.28 & -2.06 & 5.76 & -5.80 & -3.00 & 6.40 & 8.55 & 4.35 & 0.85 & 0.03 & 0 \\
-0.18 & 1.58 & -5.46 & 9.00 & -4.20 & -7.73 & 6.78 & 13.85 & 6.55 & 0.54 & 0 \\
0.11 & -1.07 & 4.25 & -8.96 & 9.24 & 0.16 & -11.47 & 6.82 & 19.26 & 5.26 & 0 \\
-0.06 & 0.62 & -2.71 & 6.60 & -9.26 & 5.66 & 4.28 & -12.48 & 9.58 & 20.11 & 0 \\
0.04 & -0.36 & 1.66 & -4.38 & 7.08 & -6.47 & 0.90 & 6.72 & -12.11 & 16.49 & 0 \\
-0.10 & 1.11 & -5.63 & 17.14 & -35.00 & 50.40 & -52.50 & 40.00 & -22.50 & 10.00 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0
\end{bmatrix}$$

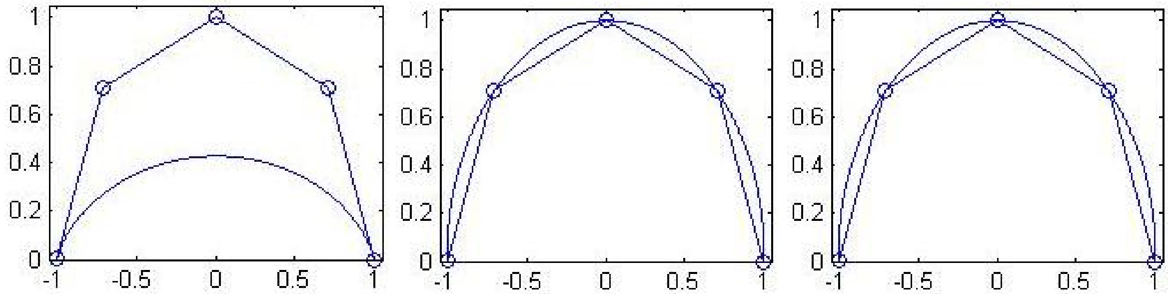
It should be noted that the classical PIA method depends on the $\rho(I - B)$, and furthermore, the initial parameterizations determine the convergence rate of the PIA method. Hence, although the classical PIA method usually have the desirable convergence rate in most cases, the convergence rate is uncertain which changes along with the initial parameterziations. The example 1 shows that another advantage of our method is that it is independent of the choice of the initial parameters, which has the fastest convergence rate, because the spectral radius $\rho(I - \overline{B}) = 0$.

In order to make a comparison, we set the precision $\varepsilon = 0.01$, and the error is measured by the formula $\varepsilon = \max_{0 \leqslant i \leqslant n}\left\{\left\|\overline{P_i} - \overline{R}_n(t_i)\right\|\right\}$. We firstly apply the classical PIA to example 1, and we find that the number of iterations required is 190. Secondly, we use the WPIA method presented by Lu, the convergence rate is accelerated nearly



(a) (b) (c)

Figure 1 Bézier basis.(a) Initial curve,(b) After 8 iterations,(c) After 11 iterations.



(d) (e) (f)

Figure 2 Wang-ball basis. (a)Initial curve, (b) After 3 iterations,(c) After 5 iterations.

two times. Finally, we use the extended PIA method: the Fig.1(a),(b),(c) shows the fitting effect at different level of iterations. We find that the error is $2.42e-007$ only after 11 iterations. When compared with the classical PIA method and WPIA method, the convergence rate of our method has been accelerated almost ten times.

At the same time, the example 2 shows that our method can also be applied to non-NTP basis, Fig.2(a),(b),(c) show that the fitting effect at different level of iterations. In fact, The Fig.2(b) shows that the desirable fitting effect have already been gained only after 3 iterations. Furthermore, we find that the error is : $1.5101e-015$. In Fig.2(c) after 5 iterations. So, for the non-NTP basis, the extended method is also quite effective. By computation, the transform matrix is

$$U=\begin{bmatrix} 0.7232 & 0.4068 & 0.1808 & 0.0452 & 0 \\ -2.4822 & 3.3870 & 2.2139 & 0.3763 & 0 \\ 0.2870 & -2.4197 & 3.4442 & 3.4077 & 0 \\ -0.5000 & 2.6667 & -6.0000 & 8.0000 & 0 \\ 0 & 0 & 0 & 0 & 1.000 \end{bmatrix}.$$

5. Conclusions

In this paper, we link the PIA method with the power series expansion of the matrix. Through introducing the transform matrix, we present a more general iterative format of PIA method which includes the classical PIA, LPIA and WPIA. In addition, we give a special form of transform matrix which can be applied to both NTP basis and any other basis functions. Compared with other form of the PIA methods, numerical examples show that our method has the fastest convergence rate with only a little computational complexity.

Acknowledgement

The research of this paper is supported by the National Nature Science Foundation of China (Nos. 60970079 and 60933008)

References

[1] D. X. Qi, Z. Tian, Y. Zhang, J. Feng, The method of numeric polish in curve fitting, Acta Mathematica Sinica, **18**, 173-184(1975).

[2] C. D. Boor. How does Agee' s smoothing method work?, ftp://ftp.cs.wisc.edu./Approx/agee.pdf

[3] H. W. Lin, H. J. Bao, C. S. Dong, Constructing iterative non-uniform B-spline curve and surface to fit data points, Science in China Series F: Information Sciences, **47**, 315-331(2004).

[4] H. W. Lin, H. J. Bao, G. J. Wang, Totally positive bases and progressive iteration approxima-tion, Computer and Mathematics with Applications, **50**, 575-586(2005).
[5] T. Maekawa, Y. Matsumoto, K. Namiki, Inter-polation by geometric algorithm. Computer Aided Design, **39**, 313-323(2007).
[6] J. Chen, G. J. Wang, Progressive iterative approximation for triangular Bzier surfaces, Computer-Aided Design, **43**, 889-895(2011).
[7] H. W. Lin, Adaptive data fitting by the progressive-iterative approximation, Computer Aided Geometric Design, **29**, 463-473(2012).
[8] H. W. Lin, Local progressive-iterative approximation format for blending curve and patches, Computer Aided Geometric Design, **27**, 322-339(2010).
[9] L. Z. Lu, Weighted progressive iteration approximation and convergence analysis, Computer Aided Geometric Design, **27**, 129-137(2010).
[10] B. Stephen, Matrix methods for engineers and scientists. McGraw-Hill, Berkshire, UK, 1979.
[11] J. Delgado, J. M. Peoa, Progressive iterative approximation and bases with the fastest convergence rates, Computer Aided Geometric Design, **24**,10-18(2007).
[12] J. Chen, G. J. Wang, Two kinds of generalized progressive iterative approximation, Atca Automatica Sinica **38**, 135-139(2012).

Shaohui Deng received the MS degree in applied mathematics from Zhejiang University in 2009, He is currently PhD Candidate in Zhejiang University. His research interests are in the areas of CAGD and CG.

Sujie Guan Sujie Guan, the research interests are computer graphics, mathematical logic and modal logic.

Guozhao Wang received the MS degree in applied mathematics from Zhejiang University in 1982, He is currently a professor and doctoral tutor in Zheji-ang University. His research interests are in the areas of CAGD and CG.

Shaobo Deng the research interests are computer graphics, Mathematical logic and modal logic. He is currently PhD Candidate in key Laboratory of intelligent information , Institute of Computing Technology ,Chinese Academy of Sciences.