


Article

New Progressive Iterative Approximation Techniques for Shepard-Type Curves

Umberto Amato ¹  and Biancamaria Della Vecchia ^{2,*} 

¹ Istituto per le Scienze Applicate ed i Sistemi Automatici ed Intelligenti, Consiglio Nazionale delle Ricerche, Via Pietro Castellino 111, 80131 Napoli, Italy; umberto.amato@cnr.it

² Dipartimento di Matematica, Università di Roma 'La Sapienza', Piazzale Aldo Moro 5, 00185 Roma, Italy

* Correspondence: biancamaria.dellavecchia@uniroma1.it

Abstract: Progressive iterative approximation (PIA) technique is an efficient and intuitive method for data fitting. In CAGD modeling, if the given data points are taken as initial control points, PIA process generates a series of shaping curves by adjusting the control points iteratively, while the limit curve interpolates the data points. Such format was used successfully for Shepard-type curves. The aim of the paper is to construct simple variants of the PIA method for Shepard-type curves producing novel curves modeling data points, so the designer can choose among several pencils of shapes outlining original control polygon. Matrix formulations, convergence results, error estimates, algorithmic formulations, critical comparisons, and numerical tests are shown. An application to a progressive modeling format by truncated wavelet transform is also presented, improving in some sense analogous process by truncated Fourier transform. By playing on two shapes handles—the number of base wavelet transform functions and the iteration level of PIA algorithm—several new contours modeling the given control points are constructed.

Keywords: Shepard-type operators; progressive iterative approximation techniques; approximation of inverse of matrix; modeling; truncated wavelet transform



Citation: Amato, U.; Della Vecchia, B.

New Progressive Iterative Approximation Techniques for Shepard-Type Curves. *Symmetry* **2022**, *14*, 398. <https://doi.org/10.3390/sym14020398>

Academic Editors: Vyacheslav Yukalov, Igor Andrianov and Simon L. Gluzman

Received: 11 January 2022

Accepted: 7 February 2022

Published: 17 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Shepard-type operators are rational, positive operators widely used in classical approximation theory and in scattered data interpolation problems (see, e.g., [1–7]). In particular, they achieve approximation results not possible by polynomials and relative pointwise and uniform direct and converse results and simultaneous approximation statements can be found, for example, in [5–8]. In [9], a modification of Shepard-type operators according to Gupta variants (based on a double summation depending on a positive parameter α) was introduced and studied. Its asymptotic behavior approaching piecewise constants made it suitable for applications to compression of piecewise images but not to CAGD. Recently, in [1], Shepard-type operators have been studied to construct Shepard-type curves useful in CAGD. Such curves overcome some of the original Shepard operator's drawbacks and have some advantages with respect to the Bézier case. A progressive iterative approximation (PIA in short) technique for Shepard-type curves and its weighted generalization (WPIA in short) were also developed in [1]. Just by adjusting the control points iteratively, PIA process presents an intuitive and straightforward way to fit data points. It generates a curve sequence with finer and finer precision and the limit of the sequence interpolates the data points. An extension to surfaces case was also investigated in [1]. Such procedure is faster than analogous PIA technique for Bézier, B-Splines, and NURBS curves (see e.g. [10,11]). Applications of PIA techniques to image super-resolution, video data compression, online handwritten synthesis, implicit curve reconstruction, fairing curve generation, local modeling, offset of planar curves, conversion of rational curves into nonrational curves, and least squares fitting are also studied [12–20].

In [1,21], generalizations of PIA technique at a quicker convergence rate, but at a higher computational cost, were introduced; however, they are not always interesting from

the modeling point of view: indeed, the designer's purpose is to draw a pencil of curves shaping the given control polygon, so a too fast sequence of curves is not always useful.

Hence, it is interesting to construct alternative PIA techniques for Shepard-type curves, at suitable convergence rate and computational cost.

The aim of this paper is to give a positive answer to this problem, introducing variants of PIA process with such properties. In Section 2, a preliminary critical revisit of PIA technique is presented, pointing out matrix formulation and properties, error expressions, computational aspects, and interesting cases. Such analysis is extended in Section 3 to four variants of PIA process based on simple iterative procedures approximating inverse of collocation matrix for Shepard-type operators at same computational cost as original PIA method; matrix formulations, convergence results, algorithmic considerations, error estimates, particular cases, and critical comparisons are shown. By acting on iteration level, like shape handle, the designer can choose among novel intermediate silhouettes between original Shepard-type curve and global interpolating Shepard-type curve at proper speed.

In Section 4 by truncated wavelet transform and PIA technique, we construct a procedure to generate Shepard-type curves useful in CAGD modeling that enhances shaping capabilities of the analogous format by truncated Fourier transform considered in [2]. We consider the number of base wavelet transform functions and the processing degree of PIA format as shape tools to draw novel curves going from Shepard-type curve to interpolating Shepard-type curve.

To prove the main results, we used the eigenstructure of Shepard-type operators, new numerical approximations of the inverse of collocation matrix and truncation wavelet transform.

Theoretical results are confirmed by numerical tests in Section 5.

2. PIA Format for Shepard-Type Curves

First, we examine the PIA technique by introducing Shepard-type curves [1]. Let $A_n(t) = [A_{n,0}(t), A_{n,1}(t), \dots, A_{n,n}(t)]^T$, with

$$A_{n,i}(t) = \frac{1}{\frac{(t-t_i)^s + \lambda}{\sum_{i=0}^n \frac{1}{(t-t_i)^s + \lambda}}}, \quad (1)$$

where $0 \leq i \leq n$, $n \in \mathbb{N}$, $t \in [0, 1]$, $t_i = \frac{i}{n}$, $i = 0, 1, \dots, n$, $0 < \lambda$, $\lambda = o(n^{-s})$ and $s > 2$ even.

If $P = [P_0, P_1, \dots, P_n]^T$, $P_i \in \mathbb{R}^d$, $i = 0, 1, \dots, n$, $d \geq 2$, is the given control vector, we define a parametric Shepard-type curve $S_n[P, t]$ as

$$S_n[P, t] = \sum_{i=0}^n A_{n,i}(t)P_i = A_n(t)P.$$

Such curves have properties that are interesting in CAGD (see, e.g., [1]): $S_n[P]$ is a rational curve that preserves points. It lies in the convex hull of the control polygon defined by P . It satisfies the pseudolocal control property—i.e., all base functions $A_{n,j}(t)$, $0 \leq j \leq n$, reach their maximum near to 1 for $t = t_j$, or equivalently each point P_j deeply affects the silhouette of the curve close to $t = t_j$.

In addition, Shepard-type curves have the property of symmetry, i.e., $S_n[P, t] = S_n[P', 1-t]$, with $P' = [P_n, P_{n-1}, \dots, P_0]^T$. The presence of λ at the denominators in (1) makes $S_n[P]$ a curve near-interpolating the control points, overcoming the flat spots drawback affecting the original Shepard operator (corresponding to $\lambda = 0$ case).

In [1], a PIA technique for Shepard-type curves was introduced and studied. Starting with an initial Shepard-type curve, PIA process constructs a sequence of fitting curves by

adjusting the control points iteratively. The limiting curve is the global Shepard-type curve interpolating the data set, defined by

$$G_n[P, t] = \sum_{k=0}^n A_{n,i}(t) Y_i, \tag{2}$$

with

$$G_n[P, t_i] = P_i, \quad i = 0, 1, \dots, n.$$

In details, given the control vector $P = [P_0, \dots, P_n]^T$ and the basis $A_{n,i}(t), i = 0, \dots, n$, defined by (1), we generate the initial curve

$$\gamma^0(t) = S_{n,0}[P, t] = S_n[P, t] = \sum_{i=0}^n A_{n,i}(t) P_i^0,$$

with $P_i^0 = P_i, i = 0, \dots, n$. Then, we calculate the remaining curves of the sequence $\gamma^{m+1}(t) = S_{n,m+1}[P, t]$, for $m \geq 0$, as follows

$$\gamma^{m+1}(t) = S_{n,m+1}[P, t] = \sum_{i=0}^n P_i^{m+1} A_{n,i}(t), \tag{3}$$

with

$$P_i^{m+1} = P_i^m + \bar{\Delta}_i^m,$$

and $\bar{\Delta}_i^m$ the adjusting vectors given by

$$\bar{\Delta}_i^m = P_i - \gamma^m(t_i), \quad i = 0, 1, \dots, n.$$

Then, the iterative process can be written in matrix form as follows:

$$\begin{aligned} [\bar{\Delta}_0^m, \bar{\Delta}_1^m, \dots, \bar{\Delta}_n^m]^T &= (I - B) [\bar{\Delta}_0^{m-1}, \bar{\Delta}_1^{m-1}, \dots, \bar{\Delta}_n^{m-1}]^T \\ &= (I - B)^{m-1} [\bar{\Delta}_0^0, \bar{\Delta}_1^0, \dots, \bar{\Delta}_n^0]^T, \end{aligned} \tag{4}$$

with B , the collocation matrix of $A_n(t)$ basis, i.e.,

$$B = \begin{pmatrix} A_{n,0}(t_0) & A_{n,1}(t_0) & \cdots & A_{n,n}(t_0) \\ A_{n,0}(t_1) & A_{n,1}(t_1) & \cdots & A_{n,n}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,0}(t_n) & A_{n,1}(t_n) & \cdots & A_{n,n}(t_n) \end{pmatrix}. \tag{5}$$

We remark that B is a positive, centrosymmetric, stochastic, diagonally dominant matrix (see [1,21]). Since B is strictly diagonally dominant, it is invertible. Since B is stochastic, we deduce that the eigenvalues of B are positive and less or equal 1, hence the spectral radius of matrix $I - B$, i.e., $\rho(I - B)$ is less than 1. Moreover, from the strict positivity of entries of stochastic matrix B , it follows that $\lim_k (B^k)_{i,j} = \pi_j$, with π , the row eigenvector associated with eigenvalue 1, which is unique.

We say that γ^m curves defined by (3) satisfy the PIA property, iff $\lim_m \gamma^m(t_i) = P_i, i = 0, \dots, n$.

Theorem 1. *Since $\rho(I - B) < 1$, curves γ^m satisfy the PIA property, i.e.,*

$$\lim_m S_{n,m}[P, t_i] = G_n[P, t_i] = P_i, \quad i = 0, \dots, n.$$

The following relations are useful in implementing above process

$$\begin{aligned}
 P_i^{m+1} &= Z_{m+1}P_i^0, \quad i = 0, \dots, n, \\
 Z_{m+1} &= \sum_{i=0}^{m+1} (I - B)^i = Z_m + (I - B)H_m, \quad Z_0 = I, \\
 H_{m+1} &= (I - B)H_m, \quad H_0 = I.
 \end{aligned}
 \tag{6}$$

Putting

$$E_m = I - BZ_m,$$

it results

$$E_{m+1} = E_m - B(I - B)^{m+1}. \tag{7}$$

Moreover,

$$E_{m+1} = (I - B)^{m+2}. \tag{8}$$

Proof. Relations (6) descend from (3) and (4).

Now, we prove (7). From (6)

$$\begin{aligned}
 E_{m+1} &= I - BZ_{m+1} = I - B \sum_{i=0}^{m+1} (I - B)^i \\
 &= I - B \sum_{i=0}^m (I - B)^i - B(I - B)^{m+1} \\
 &= I - B \left(Z_m + (I - B)^{m+1} \right) = E_m - B(I - B)^{m+1}.
 \end{aligned}$$

Now, we prove (8). From (7), we have

$$\begin{aligned}
 E_{m+1} &= E_0 - B(I - B) - B(I - B)^2 \dots - B(I - B)^{m+1} \\
 &= I - B \left(\sum_{i=0}^{m+1} (I - B)^i \right) = (I - B)^{m+2}.
 \end{aligned}$$

From $\rho(I - B) < 1$ and (8)

$$\lim_m E_m = 0, \quad \text{i.e.,} \quad \lim_m Z_m = B^{-1},$$

hence

$$\lim_m S_{n,m}[P, t_i] = B \lim_m Z_m P_i = BB^{-1}P_i = P_i, \quad i = 0, \dots, n,$$

i.e., curves γ^m satisfy the PIA property, as already proved in [1]. \square

Remark 1. We remark that at each step, a matrix multiplication is required. Moreover, we observe that the above PIA process can be interpreted in terms of classical Richardson method for systems of linear equations (see [1]).

Interesting cases are

$$\begin{aligned} \gamma^1(t) &= S_{n,1}[P, t] = \sum_{i=0}^n A_{n,i}(t)P_i^1, \\ P_i^1 &= (I + (I - B))P_i^0 = (2I - B)P_i^0, \\ \gamma^2(t) &= S_{n,2}[P, t] = \sum_{i=0}^n A_{n,i}(t)P_i^2, \\ P_i^2 &= (I + (I - B) + (I - B)^2)P_i^0 = (3I - 3B + B^2)P_i^0. \\ \gamma^3(t) &= S_{n,3}[P, t] = \sum_{i=0}^n A_{n,i}(t)P_i^3, \\ P_i^3 &= (I + (I - B) + (I - B)^2 + (I - B)^3)P_i^0 \\ &= (4I - 6B + 4B^2 - B^3)P_i^0. \end{aligned}$$

Note that the PIA process is based on well-known approximation for B^{-1}

$$B^{-1} = I + (I - B) + (I - B)^2 + \dots, \tag{9}$$

since $\rho(I - B) < 1$.

We observe that in the one-dimensional case, we find back an iterative procedure to approximate $a^{-1}, a \neq 0, |1 - a| < 1$, i.e., to solve nonlinear equation $f(x) = \frac{1}{x} - a = 0$ by a chord method, namely

$$t_{m+1} = t_m - \frac{f(t_m)}{f(t_m) - f(t_0)}(t_m - t_0) = 1 + t_m(1 - a), \quad t_0 = 1.$$

The first three iterations are

$$\begin{aligned} t_1 &= 2 - a, \\ t_2 &= 1 + (1 - a) + (1 - a)^2, \\ t_3 &= 1 + (1 - a) + (1 - a)^2 + (1 - a)^3. \end{aligned}$$

In [1,21], faster converging modifications of the above technique were introduced (at higher computational cost), so that the data points are reached in a few iterations. This result is not always interesting in CAGD modeling; indeed, from one hand, the designer’s purpose is to construct intermediate shapes outlining given control polygon; from the other hand, the original PIA process is already fast enough (cfr. [1]). Hence, it is useful to construct variants of PIA format at a suitable convergence rate and computational cost.

3. Some Variants of PIA Technique

In this section, we introduce some variants of the PIA technique.

3.1. Variant 1

Consider the following PIA-type process

$$\begin{aligned}
 \hat{S}_{n,0}[P, t] &= S_n[P, t], \\
 \hat{S}_{n,m+1}[P, t] &= \sum_{i=0}^n A_{n,i}(t)P_i^{m+1}, \quad m \geq 0, \\
 P_i^0 &= P_i, \quad i = 0, 1, \dots, n, \\
 P_i^{m+1} &= W_{m+1}P_i^0, \\
 W_m &= B \sum_{i=0}^m (i+1)(I-B)^i, \\
 W_0 &= B.
 \end{aligned}
 \tag{10}$$

Theorem 2. From $\rho(I - B) < 1$, curves defined by (10) satisfy the PIA property, that is

$$\lim_m \hat{S}_{n,m}[P, t_i] = P_i, \quad i = 0, 1, \dots, n.$$

For implementing, we use

$$\begin{aligned}
 W_m &= B\hat{W}_m = B \sum_{i=0}^m (i+1)(I-B)^i, \\
 \hat{W}_{m+1} &= \hat{W}_m + (m+2)\hat{J}_{m+1}, \\
 \hat{J}_m &= (I-B)\hat{J}_{m-1}, \\
 \hat{J}_1 &= I-B, \quad \hat{W}_0 = I.
 \end{aligned}
 \tag{11}$$

Putting

$$\hat{E}_m = I - BW_m = I - B^2\hat{W}_m,$$

one gets

$$\hat{E}_{m+1} = I - B^2 \left(\sum_{i=0}^{m+1} (i+1)(I-B)^i \right).
 \tag{12}$$

and

$$\hat{E}_{m+1} = \hat{E}_m - B^2(m+2)\hat{J}_{m+1} = \hat{E}_m - B^2(m+2)(I-B)^{m+1}.
 \tag{13}$$

Proof. Relations (11) can be obtained from (10).

Then, we prove (12) and (13). From (11),

$$\hat{E}_{m+1} = I - BW_{m+1} = I - B^2 \left(\sum_{i=0}^{m+1} (i+1)(I-B)^i \right)$$

and

$$\begin{aligned}
 \hat{E}_{m+1} &= I - BW_{m+1} = I - B^2\hat{W}_m - B^2(m+2)\hat{J}_{m+1} \\
 &= \hat{E}_m - B^2(m+2)\hat{J}_{m+1}.
 \end{aligned}$$

It is well-known that if $\rho(I - B) < 1$

$$B \sum_{i=1}^{\infty} i(I-B)^{i-1} = B \sum_{i=0}^{\infty} (i+1)(I-B)^i = B^{-1}.
 \tag{14}$$

Therefore, from (12),

$$\lim_m \hat{E}_m = 0.$$

So,

$$\lim_m W_m = B^{-1},$$

and

$$\lim_m \hat{S}_{n,m}[P, t_i] = B \lim_m W_m P_i = BB^{-1}P_i = P_i, \quad i = 0, 1, \dots, n.$$

□

Remark 2. At each step, we compute a matrix multiplication.

Comparing (7) with (13), we deduce that the above procedure is slower than the original PIA.

The first three iterations are

$$\begin{aligned} \hat{S}_{n,1}[P, t] &= \sum_{i=0}^n A_{n,i}(t)P_i^1, \\ P_i^1 &= B(I + 2(I - B))P_i^0 = B(3I - 2B)P_i^0, \\ \hat{S}_{n,2}[P, t] &= \sum_{i=0}^n A_{n,i}(t)P_i^2, \\ P_i^2 &= B(I + 2(I - B) + 3(I - B)^2)P_i^0 = B(6I - 8B + 3B^2)P_i^0, \\ \hat{S}_{n,3}[P, t] &= \sum_{i=0}^n A_{n,i}(t)P_i^3, \\ P_i^3 &= B(I + 2(I - B) + 3(I - B)^2 + 4(I - B)^3)P_i^0. \end{aligned}$$

In the one-dimensional case, we get the following iterative procedure converging to a^{-1} , $a \neq 0$, $|1 - a| < 1$,

$$\hat{t}_{m+1} = \hat{t}_m + (m + 2)a(1 - a)^{m+1}, \quad m \geq 0, \hat{t}_0 = a.$$

For example,

$$\begin{aligned} \hat{t}_1 &= a(1 + 2(1 - a)) = a(3 - 2a), \\ \hat{t}_2 &= a(1 + 2(1 - a) + 3(1 - a)^2) = a(6 - 8a + 3a^2), \\ \hat{t}_3 &= a(6 - 8a + 3a^2 + 4(1 - a)^3). \end{aligned}$$

3.2. Variant 2

Construct the following PIA-type process

$$\begin{aligned} S_{n,0}^\infty[P, t] &= S_n[P, t], \\ S_{n,m+1}^\infty[P, t] &= \sum_{i=0}^n A_{n,i}(t)P_i^{m+1}, \quad m \geq 0, \\ P_i^0 &= P_i, \quad i = 0, 1, \dots, n, \\ P_i^{m+1} &= W_{m+1}^\infty P_i^0, \\ W_m^\infty &= B^2 \sum_{i=0}^m \frac{(i + 1)(i + 2)}{2} (I - B)^i, \quad m \geq 0, \\ W_0^\infty &= B^2. \end{aligned} \tag{15}$$

Theorem 3. Since $\rho(I - B) < 1$, the curves defined by (15) satisfy the PIA property, that is

$$\lim_m S_{n,m}^\infty[P, t_i] = P_i, \quad i = 0, 1, \dots, n.$$

We implement the above process by the following relations

$$\begin{aligned}
 W_m^\boxtimes &= B^2 W_m^\circ = B^2 \sum_{i=0}^m \frac{(i+1)(i+2)}{2} (I-B)^i, \\
 W_{m+1}^\circ &= W_m^\circ + \frac{(m+2)(m+3)}{2} J_{m+1}^\boxtimes, \quad m \geq 0, \\
 J_m^\boxtimes &= (I-B) J_{m-1}^\boxtimes, \\
 J_1^\boxtimes &= I-B, \quad W_0^\circ = I.
 \end{aligned}
 \tag{16}$$

Putting

$$E_m^\boxtimes = I - BW_m^\boxtimes = I - B^3 W_m^\circ,$$

one gets

$$E_{m+1}^\boxtimes = I - B^3 \left(\sum_{i=0}^{m+1} \frac{(i+1)(i+2)}{2} (I-B)^i \right) \tag{17}$$

and

$$E_{m+1}^\boxtimes = E_m^\boxtimes - B^3 \frac{(m+2)(m+3)}{2} J_{m+1}^\boxtimes = E_m^\boxtimes - B^3 \frac{(m+2)(m+3)}{2} (I-B)^{m+1}. \tag{18}$$

Proof. From (15), one gets (16).

Now, we prove (17) and (18). By (16)

$$E_{m+1}^\boxtimes = I - BW_{m+1}^\boxtimes = I - B^3 \left(\sum_{i=0}^{m+1} \frac{(i+1)(i+2)}{2} (I-B)^i \right)$$

and

$$\begin{aligned}
 E_{m+1}^\boxtimes &= I - BW_{m+1}^\boxtimes = I - B^3 W_{m+1}^\circ \\
 &= I - B^3 W_m^\circ - B^3 \frac{(m+2)(m+3)}{2} J_{m+1}^\boxtimes \\
 &= I - BW_m^\boxtimes - B^3 \frac{(m+2)(m+3)}{2} J_{m+1}^\boxtimes \\
 &= E_m^\boxtimes - B^3 \frac{(m+2)(m+3)}{2} J_{m+1}^\boxtimes.
 \end{aligned}$$

We recall that if $\rho(I-B) < 1$

$$\sum_{i=0}^{\infty} \frac{(i+1)(i+2)}{2} (I-B^2)^i = B^{-3}.$$

Consequently, from (17)

$$\lim_m E_m^\boxtimes = 0,$$

so

$$\lim_m W_m^\boxtimes = B^{-1},$$

and

$$\lim_m S_{n,m}^\boxtimes [P, t_i] = B \lim_m W_m^\boxtimes P_i = BB^{-1} P_i = P_i, \quad i = 0, 1, \dots, n.$$

□

Remark 3. At each step, a matrix multiplication is required.

Comparing (7) and (13) with (18), we deduce that the above format is slower than original PIA technique and variant 1.

In particular,

$$\begin{aligned}
 S_{n,1}^{\infty}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^1, \\
 P_i^1 &= B^2(I + 3(I - B))P_i^0 = B^2(4I - 3B)P_i^0. \\
 S_{n,2}^{\infty}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^2, \\
 P_i^2 &= B^2(4I - 3B + 6(I - B)^2)P_i^0 = B^2(10I - 15B + 6B^2)P_i^0, \\
 S_{n,3}^{\infty}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^3, \\
 P_i^3 &= B^2(10I - 15B + 6B^2 + 10(I - B)^3)P_i^0.
 \end{aligned}$$

In the one dimensional case, one obtains the following iterative procedure converging to a^{-1} , $a \neq 0, |1 - a| < 1$,

$$t_{m+1}^{\infty} = t_m^{\infty} + \frac{(m + 1)(m + 2)}{2} a^2(1 - a)^{m-1}, \quad m \geq 0, \quad t_0^{\infty} = a^2.$$

Iterations at first, second, and third steps are given by

$$\begin{aligned}
 t_1^{\infty} &= a^2(1 + 3(1 - a)) = a^2(4 - 3a), \\
 t_2^{\infty} &= a^2(1 + 3(1 - a) + 6(1 - a)^2) = a^2(10 - 15a + 6a^2), \\
 t_3^{\infty} &= a^2(10 - 15a + 6a^2 + 10(1 - a)^3).
 \end{aligned}$$

3.3. Variant 3

Introduce the following variant of PIA process

$$\begin{aligned}
 \bar{S}_{n,0}[P, t] &= S_n[P, t], \\
 \bar{S}_{n,m+1}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^{m+1}, \quad m \geq 0, \\
 P_i^0 &= P_i, \\
 P_i^m &= T_m P_i^0, \\
 T_m &= \sum_{i=0}^m \binom{2i}{i} \frac{(I - B^2)^i}{4^i}, \\
 T_0 &= I.
 \end{aligned} \tag{19}$$

Theorem 4. Since $\rho(I - B^2) < 1$, the curves defined by (19) satisfy the PIA property, that is

$$\lim_m \bar{S}_{n,m}[P, t_i] = P_i, \quad i = 0, 1, \dots, n.$$

The following relations are useful in implementing the above format

$$\begin{aligned}
 T_m &= T_{m-1} + c_m \bar{J}_m, \\
 \bar{J}_m &= (I - B^2)^m = (I - B^2) \bar{J}_{m-1}, \\
 c_m &= \binom{2m}{m} \frac{1}{4^m} = \left(1 - \frac{1}{2m}\right) c_{m-1}, \\
 c_0 &= \binom{0}{0} \frac{1}{4^0} = 1, \\
 \bar{J}_0 &= I.
 \end{aligned}
 \tag{20}$$

Putting

$$\bar{E}_m = I - BT_m,$$

we have

$$\bar{E}_{m+1} = I - B \left(\sum_{i=0}^{m+1} \binom{2i}{i} \frac{(I - B^2)^i}{4^i} \right) \tag{21}$$

and

$$\begin{aligned}
 \bar{E}_{m+1} &= \bar{E}_m - B \binom{2(m+1)}{m+1} \frac{(I - B^2)^{m+1}}{4^{m+1}} \\
 &= \bar{E}_m - B \binom{2m}{m} \frac{2m+1}{m+1} (I - B^2)^m (I - B^2) \frac{1}{4^{m+\frac{1}{2}}}.
 \end{aligned}
 \tag{22}$$

Proof. By simple computations

$$\begin{aligned}
 c_m &= \binom{2m}{m} \frac{1}{4^m} = \frac{2m(2m-1) \dots (m+1) \dots 1}{m(m-1) \dots 1 \cdot m(m-1) \dots 1} \frac{1}{4 \cdot 4^{m-1}} \\
 &= \frac{1}{4} \frac{2m(2m-1)}{m^2} \frac{(2m-2) \dots 1}{(m-1) \dots 1 \cdot (m-1) \dots 1} \frac{1}{4^{m-1}} \\
 &= \frac{1}{4} \frac{2m(2m-1)}{m^2} \binom{2(m-1)}{m-1} \frac{1}{4^{m-1}} = \frac{1}{4} \frac{2m(2m-1)}{m^2} c_{m-1} \\
 &= \frac{2m-1}{2m} c_{m-1} = \left(1 - \frac{1}{2m}\right) c_{m-1},
 \end{aligned}$$

we deduce (20).

Then, we prove (21) and (22). From (19) and (20)

$$\bar{E}_{m+1} = I - BT_{m+1} = I - B \left(\sum_{i=0}^{m+1} \binom{2i}{i} (I - B^2)^i \frac{1}{4^i} \right),$$

and

$$\begin{aligned}
 \bar{E}_{m+1} &= I - BT_{m+1} = I - B \left(T_m + \binom{2(m+1)}{m+1} \frac{(I - B^2)^{m+1}}{4^{m+1}} \right) \\
 &= \bar{E}_m - B \binom{2(m+1)}{m+1} \frac{(I - B^2)^{m+1}}{4^{m+1}} \\
 &= \bar{E}_m - B \binom{2m}{m} \frac{2m+1}{m+1} (I - B^2)^m \frac{1}{4^{m+\frac{1}{2}}} (I - B^2).
 \end{aligned}$$

From $\rho(I - B^2) < 1$, it results

$$\sum_{i=0}^{\infty} \binom{2i}{i} (I - B^2)^i \frac{1}{4^i} = B^{-1}.$$

Therefore, from (21)

$$\lim_m \bar{E}_m = 0.$$

So

$$\lim_m T_m = B^{-1},$$

and

$$\lim_m \bar{S}_{n,m}[P, t_i] = B \lim_m T_m P_i = BB^{-1}P_i = P_i, \quad i = 0, 1, \dots, n.$$

Hence, the statement follows. \square

Remark 4. At each iteration, a matrix computation is needed.

From Stirling approximation of binomial

$$m! \sim \sqrt{2\pi m} \left(\frac{m}{e}\right)^m,$$

we deduce

$$\binom{2(m+1)}{m+1} (I - B^2)^{m+1} \frac{1}{4^{m+1}} \sim m^{\frac{3}{2}} (I - B^2)^{m+1}, \tag{23}$$

which vanishes slower than $(I - B)^{m+1}$ (cfr. (7)), hence from (22) and (23) \bar{E}_m goes to 0 slower than E_m . Comparing (22) and (23) with (13), it follows that for $m = 0, 1, 2, 3$, interesting cases in applications, $|\bar{E}_{m+1}|$ is smaller than $|\hat{E}_{m+1}|$, but asymptotically is bigger, hence for $m = 0, 1, 2, 3$, $\bar{S}_{n,m}$ is faster than $\hat{S}_{n,m}$, but asymptotically is slower. Consequently, for $m = 0, 1, 2, 3$, variant 3 gives intermediate curves between PIA and variant 1 process. Comparing (22) and (23) with (18), \bar{E}_m vanishes asymptotically slower than E_m^{\triangleleft} .

As particular cases, we find

$$\begin{aligned} \bar{S}_{n,1}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^1, \\ P_i^1 &= \left(I + \frac{1}{2}(I - B^2)\right) P_i^0 = \left(\frac{3}{2}I - \frac{1}{2}B^2\right) P_i^0, \\ \bar{S}_{n,2}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^2, \\ P_i^2 &= \left(\frac{3}{2}I - \frac{1}{2}B^2 + \frac{3}{8}(I - B^2)^2\right) P_i^0 = \left(\frac{15}{8}I - \frac{5}{4}B^2 + \frac{3}{8}B^4\right) P_i^0. \\ \bar{S}_{n,3}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_i^3, \\ P_i^3 &= \left(\frac{15}{8}I - \frac{5}{4}B^2 + \frac{3}{8}B^4 + \frac{5}{16}(I - B^2)^3\right) P_i^0. \end{aligned}$$

In the one-dimensional case, we have the following iterative procedure converging to a^{-1} , $a \neq 0$, $|1 - a^2| < 1$,

$$\bar{t}_{m+1} = \bar{t}_m + \binom{2m}{m} \frac{2m+1}{m+1} (1 - a^2)^{m+1} \frac{1}{4^{m+\frac{1}{2}}}, \quad m \geq 0, \bar{t}_0 = 1,$$

For example,

$$\begin{aligned} \bar{t}_1 &= 1 + \frac{1}{2}(1 - a^2) = \frac{3}{2} - \frac{a^2}{2}, \\ \bar{t}_2 &= \frac{3}{2} - \frac{a^2}{2} + \frac{3}{8}(1 - a^2)^2 = \frac{15}{8} - \frac{5}{4}a^2 + \frac{3}{8}a^4, \\ \bar{t}_3 &= \frac{15}{8} - \frac{5}{4}a^2 + \frac{3}{8}a^4 + \frac{5}{16}(1 - a^2)^3. \end{aligned}$$

3.4. Variant 4

Letting $q \in (0, 1)$ define the following variant of PIA process,

$$\begin{aligned}
 S_{n,0,q}^{\oplus}[P, t] &= S_n[P, t], \\
 S_{n,m+1,q}^{\oplus}[P, t] &= \sum_{i=0}^n A_{n,i}(t) P_{i,q}^{m+1}, \quad m \geq 0, \\
 P_{i,q}^0 &= P_i, \\
 P_{i,q}^m &= H_{m,q} P_i^0, \\
 H_{m,q} &= (I - q(I - B)) \sum_{i=0}^m \frac{1 - q^{i+1}}{1 - q} (I - B)^i, \\
 H_{0,q} &= I - q(I - B).
 \end{aligned}
 \tag{24}$$

Theorem 5. Since $\rho(I - B) < 1$, the curves defined by (24) satisfy the PIA property, that is,

$$\lim_m S_{n,m,q}^{\oplus}[P, t_i] = P_i, \quad i = 0, 1, \dots, n.$$

We implement the above process with the following relations

$$\begin{aligned}
 H_{m,q} &= H_{m-1,q} + (I - q(I - B))N_{m,q}, \\
 N_{m,q} &= \frac{1 - q^{m+1}}{1 - q} J_m, \\
 J_m &= (I - B)J_{m-1}, \\
 J_0 &= I.
 \end{aligned}
 \tag{25}$$

Putting

$$E_{m,q}^{\oplus} = I - BH_{m,q},$$

we have

$$E_{m+1,q}^{\oplus} = I - B(I - q(I - B)) \sum_{k=0}^{m+1} \frac{1 - q^{k+1}}{1 - q} (I - B)^k
 \tag{26}$$

and

$$E_{m+1,q}^{\oplus} = E_{m,q}^{\oplus} - B(I - q(I - B)) \frac{1 - q^{m+2}}{1 - q} (I - B)^{m+1}.
 \tag{27}$$

Proof. From (24) working as above, one gets (25)–(27).

We recall the q -binomial theorem

$$z^{-1} = (1 - q(1 - z)) \sum_{i=0}^{\infty} \frac{1 - q^{i+1}}{1 - q} (1 - z)^i, \quad |z| < 1,$$

from which we deduce if $\rho(I - B) < 1$,

$$B^{-1} = (I - q(I - B)) \sum_{i=0}^{\infty} \frac{1 - q^{i+1}}{1 - q} (I - B)^i.
 \tag{28}$$

In particular, if $q = 0$ and $q = 1$, we find back (9) and (14), respectively, which underlie PIA and variant 1 processes, respectively.

Therefore, from (26)

$$\lim_m E_{m,q}^{\oplus} = 0.$$

So,

$$\lim_m H_{m,q} = B^{-1},$$

and

$$\lim_m S_{n,m}^\oplus q[P, t_i] = B \lim_m H_{m,q} P_i = BB^{-1} P_i = P_i, \quad i = 0, 1, \dots, n.$$

and the theorem is proved. \square

Remark 5. At each step, a matrix multiplication is needed.

In the limiting cases $q = 0$ and $q = 1$, we find back PIA process $S_{n,m}[P, t]$ and variant 1 method $\hat{S}_{n,m}[P, t]$ (cfr. (6) and (10), respectively; see also (28)). Hence, parameter q can be considered as a further shape handle, bringing variant 4 from PIA process to variant 1; so, by playing on iteration level m and parameter q , the designer has at his disposal new intermediate curves gradually outlining the given control polygon.

From q -binomial theorem (see [22] for applications of q -calculus in approximation theory)

$$\prod_{i=0}^2 \frac{1}{(1 - q^i t)} = \sum_{i=0}^{\infty} \binom{n+i-1}{i}_q t^i, \quad |t| < 1,$$

with

$$\binom{m}{r}_q = \frac{(1 - q^m)(1 - q^{m-1}) \dots (1 - q^{m-r+1})}{(1 - q)(1 - q^2) \dots (1 - q^r)},$$

following the proof of Theorem 5, we can construct an analogous variant, generalizing PIA process and variant 2. We omit details.

Particular cases are

$$S_{n,1,q}^\oplus [P, t] = \sum_{i=0}^n A_{n,i}(t) P_{i,q}^1,$$

$$P_{i,q}^1 = (I - q(I - B))(I + (1 + q)(I - B)) P_i^0,$$

$$S_{n,2,q}^\oplus [P, t] = \sum_{i=0}^n A_{n,i}(t) P_{i,q}^2,$$

$$P_{i,q}^2 = (I - q(I - B)) \left(I + (1 + q)(I - B) + (1 + q + q^2)(I - B)^2 \right) P_i^0,$$

$$S_{n,3,q}^\oplus [P, t] = \sum_{i=0}^n A_{n,i}(t) P_{i,q}^3,$$

$$P_{i,q}^3 = (I - q(I - B)) \left(I + (1 + q)(I - B) + (1 + q + q^2)(I - B)^2 + (1 + q + q^2 + q^3)(I - B)^3 \right) P_i^0.$$

In the one-dimensional case, we find the following iterative procedure converging to a^{-1} , $a \neq 0, |1 - a| < 1$,

$$t_{m+1}^\oplus = t_m^\oplus + (1 - q(1 - a))(1 - a) \frac{1 - q^{m+2}}{1 - q}, \quad m \geq 0,$$

$$t_0^\oplus = 1 - q(1 - a).$$

The first three iterations are

$$\begin{aligned}
 t_1^\oplus &= (1 - q(1 - a))(1 + (1 + q)(1 - a)), \\
 t_2^\oplus &= (1 - q(1 - a))\left(1 + (1 + q)(1 - a) + (1 + q + q^2)(1 - a)^2\right), \\
 t_3^\oplus &= (1 - q(1 - a))\left(1 + (1 + q)(1 - a) + (1 + q + q^2)(1 - a)^2 \right. \\
 &\quad \left. + (1 + q + q^2 + q^3)(1 - a)^3\right).
 \end{aligned}$$

4. A PIA-Type Technique via Truncated Wavelet Transform

In [2], a method to construct new Shepard-type curves with good shaping behaviour was introduced by PIA algorithm and truncated DFT. The number of base Fourier functions and the iteration level of PIA algorithm handle modeling the outline of Shepard-type curve, getting as a limit case the Shepard-type interpolating curve given by (2). The truncation procedure was made in analogy with truncation occurring in some statistical contexts involving Fourier transform.

Here, we extend such technique to wavelet transform. As a representation of a function, we consider an orthonormal basis of functions obtained from dilations and translations of compactly supported and periodic scaling ($\phi(t)$) and wavelet ($\psi(t)$) functions:

$$\phi_{j_0k}(t), k = 0, \dots, 2^{j_0-1}; \psi_{jk}(t), j > j_0, k = 0, \dots, 2^{j-1},$$

where

$$\phi_{jk}(t) = 2^{j/2}\phi(2^j t - k), \psi_{jk}(t) = 2^{j/2}\psi(2^j t - k)$$

and j_0 is a reference level.

As a well-known method (e.g., [23]), wavelet transform has excellent properties from the approximation point of view for wide classes of functions. In addition, it turns to have a sparse representation, in the sense that its coefficients quickly drop to 0. Therefore, a limited set of coefficients could be sufficient to accurately represent the function to be approximated. Further, orthogonality of the transform makes it very easy to compute inverse transforms. Finally, availability of a multiresolution algorithm for discrete data allows one to compute the discrete wavelet transform very quickly with a computational cost of $O(n)$ operations. In this case of discrete data, the discrete wavelet transform, z^w of a set of dyadic data z of length n can be expressed as $z^w = Wz$, with W being the wavelet matrix of order n .

Therefore, the idea is to compute the wavelet transform of the discrete points $B^{-1}P$ and, before inverse transforming them, to threshold them with a decreasing threshold, retaining only coefficients higher than the threshold (in absolute value). Depending on the values of the threshold, a family of curves is obtained approximating the data.

We sketch the procedure. From the global interpolating Shepard-type curve Formula (2) and by (5)

$$BY = P, \quad Y = [Y_0, Y_1, \dots, Y_n]^T,$$

or equivalently

$$Y = B^{-1}P = VP, \quad V = B^{-1}. \tag{29}$$

From (2) and (29), putting $A_n(t) = A$,

$$G_n[P, t] = A \cdot Y = AW^T WY = AW^T WVP,$$

with W the wavelet transform matrix giving back the wavelet coefficients $P^w = WVP$. Now let us retain only the highest $k, k \leq n$, wavelet coefficients P^w (in absolute value). This corresponds to select only k bases of the transform corresponding to the selected wavelet coefficients. Equivalently, in matrix notation we can consider truncated wavelet matrices $W^{(k)} \in \mathbb{R}^{k \times n+1}$ that include only rows corresponding to the retained wavelet coefficients. Analogously to the FT format introduced in [2], such truncation procedure was inspired by some statistical problems involving wavelets.

Then, we introduce the truncated interpolation curve of Shepard-type:

$$G_n^{(k)}[P, t] = A_n(t) \left(W^{(k)} \right)^T W^{(k)} V P. \tag{30}$$

From (30) it follows that, when $k = n$, $\left(W^{(n)} \right)^T W^{(n)} = W^T W = I$ and we deduce (2). Therefore, when gradually $0 \leq k \leq n$ in (30) we get different curves going closer and closer to the interpolating Shepard-type curve.

From (30) and above PIA format, we deduce a technique constructing a pencil of curves of Shepard-type outlining the given data points. We summarize the process. In (30), matrix V is replaced by

$$V_m = \sum_{k=0}^m (I - B)^k, \tag{31}$$

with m being the iteration level of PIA procedure. From (9) $\lim_m V_m = V$; because of fast convergence, with a few iterations V_m can get close to V .

So, if the designer plays with the number k of basis wavelet functions in (30) and the processing degree m of the PIA algorithm in (31), intermediate shapes not attainable by PIA procedure are drawn.

5. Implementation and Examples

Here, we examine some tests to show the effectiveness of our algorithms.

5.1. Example 1

Consider a helix of radius 5 given by (see, e.g., [1])

$$(x(t), y(t), z(t)) = (5 \cos t, 5 \sin t, t), \quad t \in [0, 6\pi].$$

A sequence of 19 control points $P_i, i = 0, \dots, 18$, is sampled from the helix as

$$(x(s_i), y(s_i), z(s_i)), \quad s_i = \frac{\pi}{3}i, \quad i = 0, 1, \dots, 18. \tag{32}$$

We construct PIA curves given by (6) based on above 19 initial control points (32) with $s = 4, \lambda = 4 \cdot 10^{-6}$ (see (1)). In Figure 1 one can see the PIA Shepard-type curves at the initial step and after 1, 2 and 3 iterations. From Figure 1 we deduce that the curve sequence defined by PIA format converges fast to the interpolating Shepard-type curve, as expected from Theorem 1 in Section 2.

5.2. Example 2

We examine the above helix example for variants of PIA process. First construct curves defined by format (10). Figure 2 shows the corresponding curves at the initial step and after 1, 2 and 3 iterations. As the iteration level increases, the modeling power of technique (10) is evident. The curves approach the given control points at a slower convergence rate than Example 1, as remarked in Section 3.1.

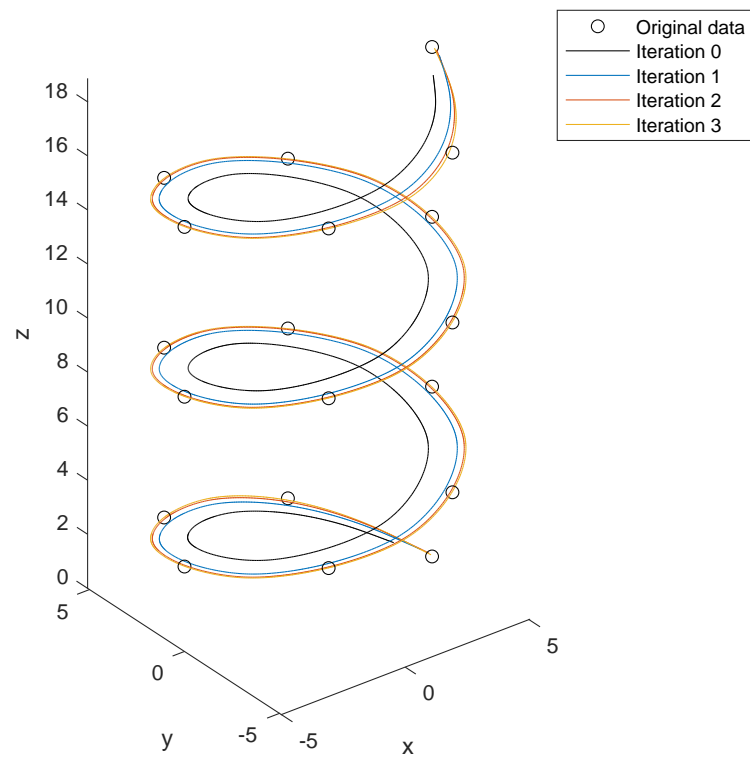


Figure 1. PIA curves at iteration levels 0, 1, 2 and 3 for the Example 1.

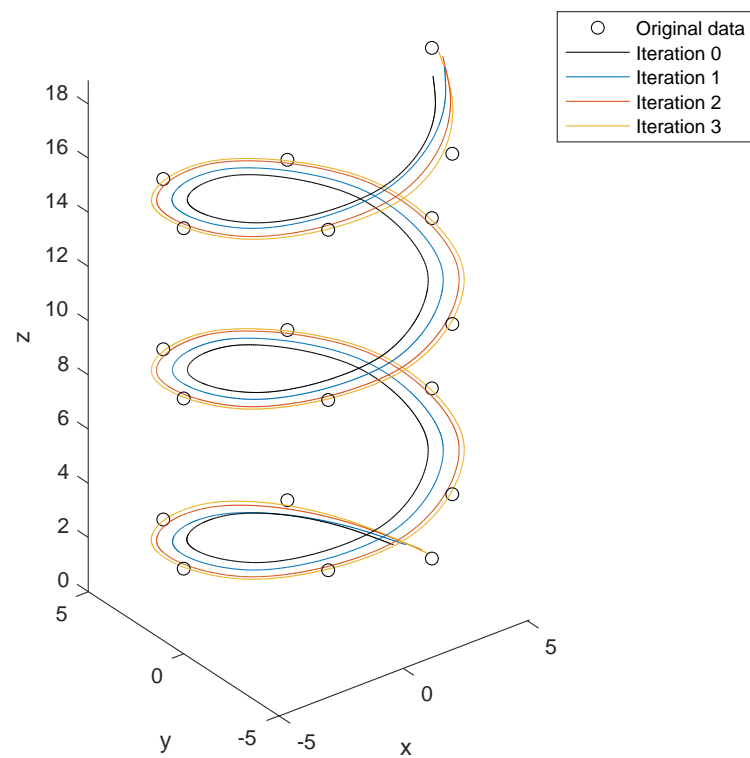


Figure 2. Variant 1 curves at iteration levels 0, 1, 2 and 3 for the Example 2.

5.3. Example 3

We compare the same helix example for variant 2 of PIA process (15). In Figure 3 corresponding curves at the initial step and after 1, 2, and 3 iterations are presented. When we proceed with the iterations, the shaping behaviour of format (15) is pointing out. The intermediate curves model the helix more slowly than Examples 1 and 2, as observed in Section 3.2.

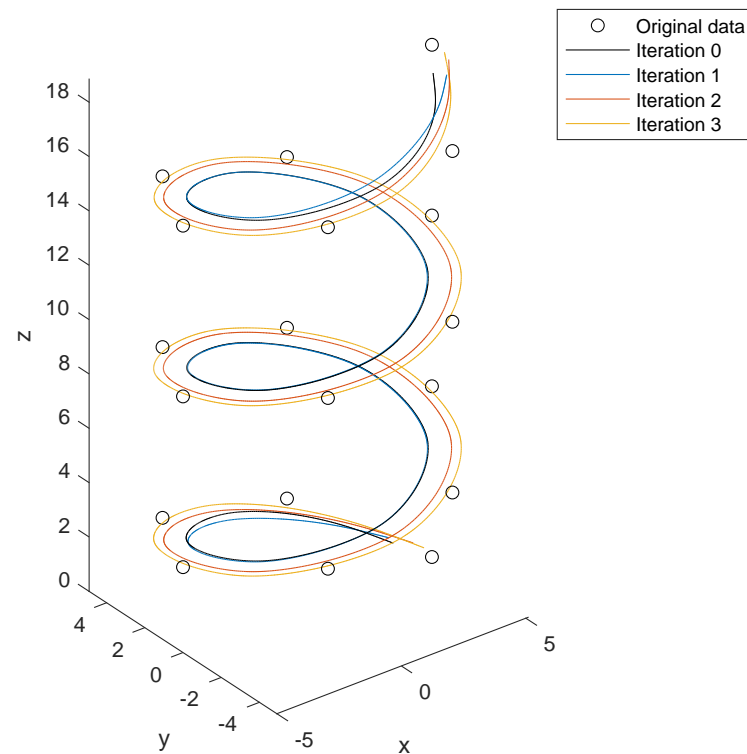


Figure 3. Variant 2 curves at iteration levels 0, 1, 2 and 3 for the Example 3.

5.4. Example 4

Consider curves defined by technique (19) for above helix test. In Figure 4 one can see the corresponding curves at the initial step and after 1, 2 and 3 iterations and one can choose among several shapes modeling the given control polygon. As remarked in Section 3.3, here approximating curves represent an alternative to previous formats.

5.5. Example 5

We study the same helix example for variant 4 of PIA process (24). Figure 5 shows the corresponding curves for $m = 1$ and $q = 0, 1/3, 2/3, 1$, respectively. From Figure 5 we can see the shaping capability of variant 4: indeed, if $q = 0$ and $q = 1$, we find back the curves obtained by the first two iterations of PIA and variant 1 algorithms (see Section 3.4), respectively, while the choices $q = 1/3, 2/3$ give intermediate curves approximating the helix.

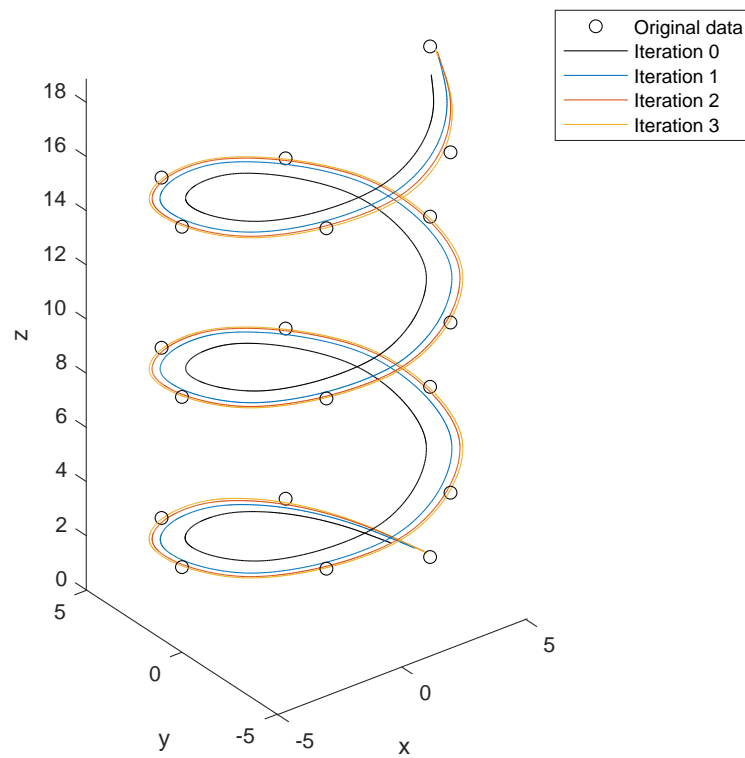


Figure 4. Variant 3 curves at iteration levels 0, 1, 2 and 3 for the Example 4.

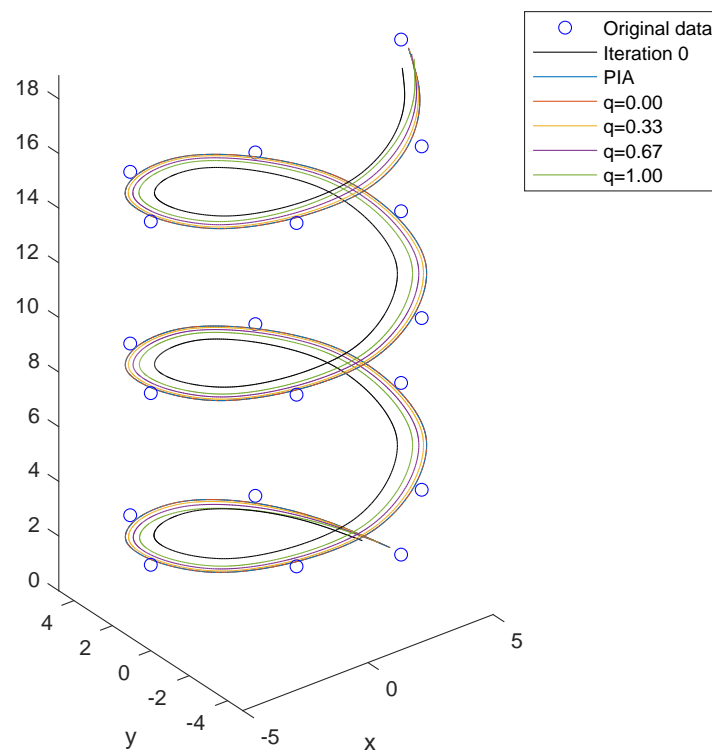


Figure 5. Variant 4 curves at iteration level 1 for $q = 0, 1/3, 2/3, 1$ for the Example 5.

5.6. Example 6

Consider Archimedes spiral given by

$$(x(t), y(t)) = (t \cos 6\pi t, t \sin 6\pi t), t \in [0, 1].$$

We extract from the spiral a sample of 16 ($n = 15$) control points as

$$(x(s_i), y(s_i)), s_i = \frac{i}{n}, i = 0, \dots, n.$$

They form the control polygon in Figure 6 in black color. We consider the curves obtained by PIA-type technique via truncated wavelet transform of Section 4. To this purpose a Symmlet basis was used with 10 vanishing moments and reference level $j_0 = 0$. Moreover we consider one PIA iteration ($m = 1$) in (31), with $s = 4$ and $\lambda = 10^{-5}$. Starting from the above control points, Figure 6 shows the fit by curves defined by the above format retaining from 7 to all the 16 wavelet coefficients. A comparison with the analogous curves obtained by FT on the same example in [2] shows the enhanced capability of PIA-type technique via truncated wavelet transform to construct a higher number of different shaping curves.

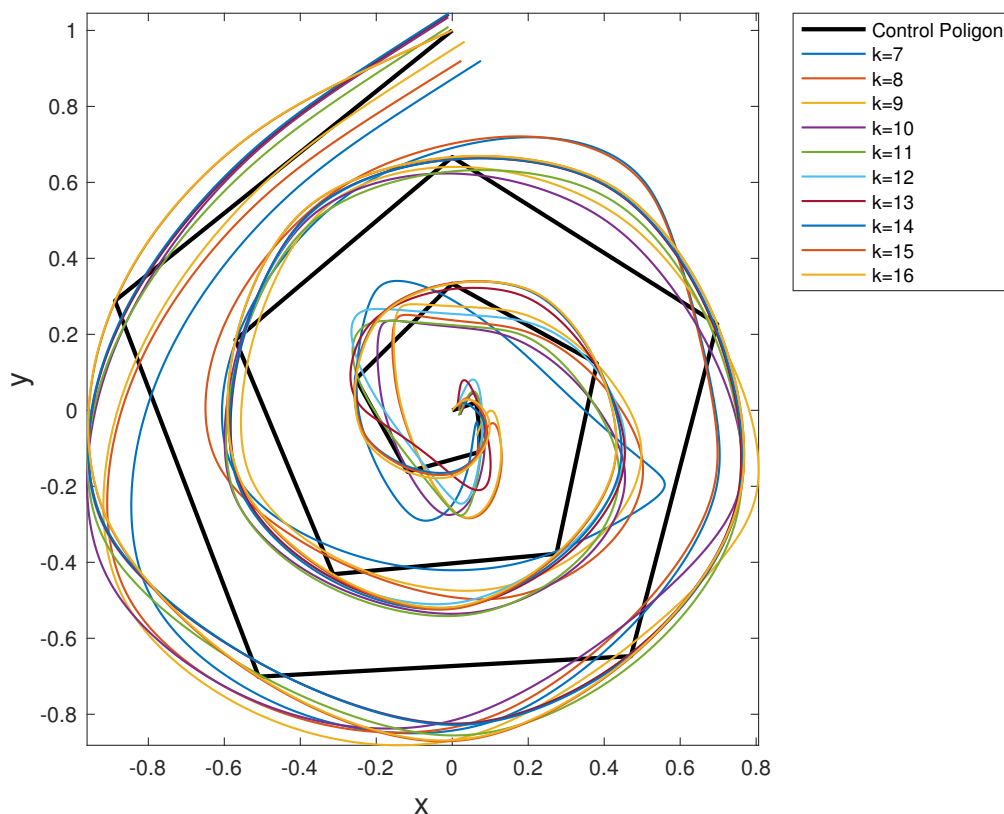


Figure 6. PIA-type curves via truncated wavelet transform with one PIA iteration and a number of retained wavelet coefficients ranging from 7 to 16 for the Archimedes spiral.

6. Conclusions

In this paper four variants of Progressive Iterative Approximation method for Shepard-type curves introduced in [1] are developed, whose limit is the Shepard-type curve interpolating the data points. Such variants are based on simple iterative procedures approximating inverse of collocation matrix of Shepard-type operators. They draw novel intermediate pencils of curves between original Shepard-type curve and global interpolating Shepard-type curves, at the same computational cost as original PIA process. So the designer has at his disposal several choices to model given control polygon. Critical analysis of such

algorithms, matrix formulations, convergence results, error estimates, particular cases, comparisons and numerical examples are discussed. A CAGD modeling technique for Shepard-type curves by truncated wavelet transform and PIA algorithm is also developed at higher performance than analogous truncated Fourier transform format.

The high degree of Shepard-type operator represents a drawback and modification of Shepard-type operator at a lower degree will be the subject of further work.

Author Contributions: Conceptualization, B.D.V.; methodology, B.D.V. and U.A.; software, U.A.; investigation, B.D.V.; writing—original draft preparation, B.D.V.; writing—review and editing, U.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Notation

The following notation is used in this manuscript:

$[x_1, x_2, \dots, x_N]^T$	Transposed vector
$S_n[P, t]$	Parametric curve relative to control vector P , with variable t
\hat{X}	Variant 1 applied to X
$X^{\times 2}$	Variant 2 applied to X
\bar{X}	Variant 3 applied to X
X^{\oplus}	Variant 4 applied to X
$a \sim b$	iff $ a/b \leq C$, with C positive constant independent of a and b
$W^{(k)}$	truncation applied to W

References

- Amato, U.; Della Vecchia, B. Modelling by Shepard-type curves and surfaces. *J. Comput. Math. Appl.* **2016**, *20*, 611–634.
- Amato, U.; Della Vecchia, B. Rational Approximation on Exponential Meshes. *Symmetry* **2020**, *12*, 1999. [\[CrossRef\]](#)
- Allasia, G. A class of interpolator positive linear operators: Theoretical and computational aspects. In *Approximation Theory, Wavelets and Applications*; NATO ASI Series C; Singh, S.P., Ed.; Kluwer: Dordrecht, The Netherlands, 1995; Volume 454, pp. 1–36.
- Anjyo, K.; Lewis, J.P.; Pighin, F. Scattered Data Interpolation for Computer Graphics. In *ACM SIGGRAPH 2014 Courses*; Association for Computing Machinery: New York, NY, USA, 2014.
- Della Vecchia, B. Direct and converse results by rational operators. *Constr. Approx.* **1996**, *12*, 271. [\[CrossRef\]](#)
- Della Vecchia, B.; Mastroianni, G. Pointwise simultaneous approximation by rational operators. *J. Approx. Theory* **1991**, *65*, 140–150. [\[CrossRef\]](#)
- Szabados, J. On a problem of R. Devore. *Acta Math. Acad. Sci. Hung.* **1976**, *27*, 219–223. [\[CrossRef\]](#)
- Vértesi, P. Saturation of the shepard operator. *Acta Math. Hung.* **1996**, *72*, 307–317. [\[CrossRef\]](#)
- Amato, U.; Della Vecchia, B. On Shepard–Gupta-type operators. *J. Inequ. Appl.* **2018**, *2018*, 232. [\[CrossRef\]](#) [\[PubMed\]](#)
- Carnicer, J.M.; Delgado, J.; Peña, J.M. On the Progressive Iteration Approximation Property and Alternative Iterations. *Comput. Aided Geom. Des.* **2011**, *28*, 523–526. [\[CrossRef\]](#)
- Lin, H.W.; Bao, H.J.; Wang, G.J. Totally Positive Bases and Progressive Iteration Approximation. *Comput. Math. Appl.* **2005**, *50*, 575–586. [\[CrossRef\]](#)
- Bezine, H.; Alimi, A.M. Handwriting perceptual classification and synthesis using discriminate HMMs and progressive iterative approximation. *Neural Comput. Appl.* **2020**, *32*, 16549–16570. [\[CrossRef\]](#)
- Chantakamo, A.; Dejdumrong, N. The Analysis of PIA Algorithm Efficiency in Converting Rational Bézier Curves into Corresponding Non-rational Curves. In Proceedings of the 2014 11th International Conference on Computer Graphics, Imaging and Visualization, Singapore, 6–8 August 2014; pp. 52–56.
- Ebadi, M.; Ebrahimi, A. Video Data Compression by Progressive Iterative Approximation. *Int. J. Interact. Multimed. Artif. Intell.* **2021**, *6*, 189–195. [\[CrossRef\]](#)
- Hamza, Y.F.; Lin, H.; Li, Z. Implicit progressive-iterative approximation for curve and surface reconstruction. *Comput. Aided Geom. Des.* **2020**, *77*, 101817. [\[CrossRef\]](#)
- Lin, H. Local Progressive-Iterative Approximation Format for Blending Curves and Patches. *Comput. Aided Geom. Des.* **2010**, *27*, 322–339. [\[CrossRef\]](#)
- Lin, H.; Zhao, Y. Variational Progressive-Iterative Approximation for Fairing Curve and Surface Generation. In Proceedings of the 2011 12th International Conference on Computer-Aided Design and Computer Graphics, Jinan, China, 15–17 September 2011; IEEE Computer Society: Los Alamitos, CA, USA, 2011; pp. 258–261.
- Wang, H. On extended progressive and iterative approximation for least squares fitting. *Vis. Comput.* **2021**, 1–12. [\[CrossRef\]](#)

19. Zhang, L.; Wang, H.; Li, Y.; Tan, J. A progressive iterative approximation method in offset approximation. *J. Comput. Aided Des. Comput. Graph.* **2014**, *26*, 1646–1653.
20. Zhang, Y.; Wang, P.; Bao, F.; Yao, X.; Zhang, C.; Lin, H. A Single-Image Super-Resolution Method Based on Progressive-Iterative Approximation. *IEEE Trans. Multimed.* **2020**, *22*, 1407–1422. [[CrossRef](#)]
21. Amato, U.; Della Vecchia, B. Weighting Shepard-type operators. *Comput. Appl. Math.* **2017**, *36*, 885–902. [[CrossRef](#)]
22. Aral, A.; Gupta, V.; Agarwal, R.P. *Applications of q -Calculus in Operator Theory*; Springer: Berlin/Heidelberg, Germany, 2013.
23. Mallat, S. *A Wavelet Tour of Signal Processing. The Sparse Way*, 3rd ed.; Academic Press: Burlington, MA, USA, 2008.