# A Study on Iterative Scheme for Least Squares Progressive Iterative Approximation


Nitisha Pandey

Reg.no.: 24PHD10018

Under the guidance of

**Dr. Reena Jain**

Associate Professor

School of Advanced Sciences and Languages
Vellore Institute of Technology Bhopal, University
Kothrikalan,Madhya Pradesh-466114

# Respected Members of the Meeting

| | | |
|---|---|---|
| Dr.Hemant Kumar Nashine | Chairman | Professor and Dean, VIT Bhopal University |
| Dr. Reena Jain | Supervisor | Associate Professor, VIT Bhopal University |
| Dr. Debasisha Mishra | External Member | Associate Professor, NIT Raipur |
| Dr. Santosh Kumar Bhal | Internal Member | Assistant Professor, VIT Bhopal University |

# Outline

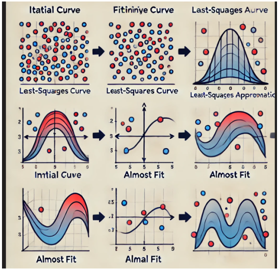
- Iterative methods
- Types of least squares method
- History of Spline Curves
- Types of Spline
- Cubic B-spline
- Curve
- Literature Review
- Road Map
- References

# Iterative methods

Iterative methods are often used when direct solutions to polynomial approximation problems are either too complex or computationally expensive. In such cases, iterative methods can efficiently provide solutions that converge toward the best approximation of a given function.

# Why we use LSPIA ?

Data fitting is a fundamental research problem in many scientific fields. The progressive iterative approximation for least squares fitting is an effective method for fitting many data points, because of its simple iterative algorithm and intuitive operation in CAD/CAM.




However, the classic LSPIA method for blending curves and surfaces has a slow convergence rate and takes a long CPU execution time.So we use the extension of LSPIA for overcoming these limitation.

# Types of Iterative Scheme

| Iterative Scheme | Convergence Rate | Advantages |
|---|---|---|
| Standard LSPIA | Linear convergence | Simple to implement, basic refinement of control points |
| Weighted LSPIA | Faster than standard LSPIA | Improves accuracy for important data points, faster convergence |
| HPLSPIA | Faster than SLSPIA and WLSPIA | More accurate approximations due to derivative info |

# LSPIA

- **Least Squares Progressive Iterative Approximation (LSPIA)** method is a fundamental technique in data analysis, widely used to find the best-fitting line or curve for a set of data points.
  **Error** $= \sum_{i=1}^{n} (y_i - C(t_i))^2$
  where,
  $C(t_i) =$ B-spline curve at parameter value $t_i$
  $y_i =$observed value.
  By minimizing this error, the B-spline curve provides a smooth and optimal approximation to the data points.
- **Weighted LSPIA**, the standard Least-Squares Polynomial Approximation is modified by assigning different weights to each data point. This allows some data points to have more influence on the approximation than others, which can be useful when certain data points are more reliable or important.

$$\mathbf{E} = \sum_{i=1}^{m} w_i \left(y_i - C_n(t_i)\right)^2$$

where,
$w_i$ are the weights assigned to each data point $x_i$.
$y_i$ is the actual value at point.
$c_n(t_i)$is the polynomial approximation at point $E$ is the weighted sum of squared differences between the actual and approximated values.

- **Hyperpower LSPIA** is an extension of LSPIA where the approximation is applied not only to the function values but also to higher-order derivatives of the function. This leads to a smoother approximation that not only fits the data points but also ensures that the polynomial's slope and curvature are close to those of the actual function.

# History of Spline Curves

Spline curves have a rich history, dating back to the early days of mechanical drafting. These curves were originally created by bending flexible strips of wood or metal, called splines.

- **Early Beginnings** The earliest splines were physical tools used by draftsmen for drawing smooth curves.
- **Mathematical Representation** In the 1960s, mathematicians **Pierre Bézier** developed mathematical formulas to represent spline curves, making them suitable for computer processing.
- **Computer Aided Design** The arrival of computers changed how splines were used, making it easier and faster to create and work with curves in many different applications.

# Spline

A **spline** is a mathematical function used to create a smooth curve through a set of points. It's constructed from multiple polynomial segments connected at certain points, known as **knots**. Spline functions are widely used in numerical analysis, **computer graphics**, data interpolation, and **curve fitting** because they allow for smooth transitions between data points while maintaining local control over the curve shape.

The **degree** of the spline indicates the degree of the polynomial used in each segment between knots.

# Types of Spline

## Comparison of Spline Types

### Linear Spline

Connects data points with straight lines, resulting in a piecewise linear function. Simple but lacks smoothness.

### Quadratic Spline

Uses quadratic polynomials to connect points, creating smoother curves than linear splines. Still lacks flexibility.

### B-Cubic Spline

Employs cubic polynomials to achieve even smoother curves with greater control over shape and flexibility. Widely used in applications.

## Continuity and Smoothness of Cubic B-Spline Curves

Cubic B-Splines are known for their smooth and continuous curves, with different levels of continuity defined by the number of continuous derivatives.

| | |
|---|---|
| C0 | Position continuity |
| C1 | Position and tangent continuity |
| C2 | Position, tangent, and curvature continuity |

# What's Special About Splines?

- Cubic Splines are function approximations that are continuous at merging points.
- They also have continuous first and second derivatives where they join.

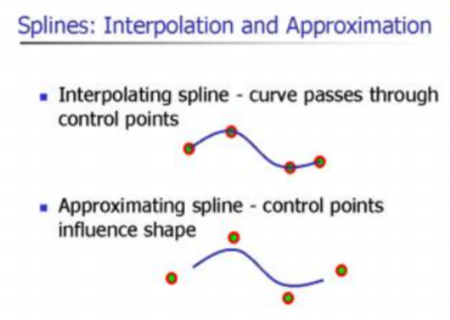

- Natural Cubic Spline is a cubic spline that has second derivative equal to zero at end points.

# Cubic B-spline

Cubic B-Splines are piecewise polynomial curves defined by a set of control points and a knot vector.They possess properties like local control, smoothness, and affine invariance, making them suitable for various applications.

- **Local Control** Moving a control point only affects a small portion of the curve, making it easy to adjust shapes precisely.
- **Smoothness** Cubic B-Splines are continuously differentiable, resulting in smooth and aesthetically pleasing curves.
- **Affine Invariance**-Transformations applied to control points are reflected in the corresponding transformations of the curve, simplifying geometric operations.

# Curve




Spline

Interpolating splines ensure exact data fitting, while approximating splines provide a smoother, more stable fit that balances the control points without passing through them.

# Curve Fitting: Overview and Types

Curve fitting is a statistical technique used to create a mathematical function (curve) that best represents a set of data points. It involves minimizing the differences between observed values and the values predicted by the curve.

| Types | Description | Form |
|---|---|---|
| Linear Curve | Models linear relationships. | $y = mx + b$ |
| Polynomial Curve | Represents complex, nonlinear relationships. | $y = a^n x^n + a^n - 1x^n - 1 + .. + a_0$ |
| Exponential Curve | A curve that increases or decreases exponentially, useful for modeling growth or decay processes. | $y = e^{ax}$ |
| Spline Curve | Piecewise polynomial that provides flexible and smooth curves | Various forms, typically cubic splines |

# Curve Representation

A cubic B-Spline curve is defined by a linear combination of basis functions, called **B-spline basis functions**, multiplied by the control points. Each basis function is a piecewise polynomial of degree three

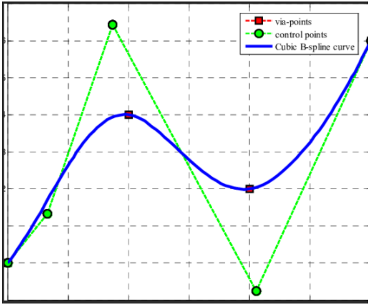
$C(t) = \sum_{i=0}^{n} B_{i,k}(t) P_i$

where

$C(t)$is the curve at parameter

$B_{i,k}(t)$ are the B-spline basis functions of order k

$P_i$ are the control points.



B-spline basis functions are defined recursively as follows:
**Order k basis functions:**

$$B_{i,k}(t) = \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1}, \\ 0 & \text{otherwise.} \end{cases}$$

**Higher degree basis functions:**

$$B_{i,k}(t) = \frac{t - t_i}{t_{i+k} - t_i} B_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(t)$$

# Types of B-spline

| Type of B-spline | Degree | Applications |
|---|---|---|
| Cubic B-spline | 3 | Computer graphics, smooth curve |
| Uniform B-spline | Varies | Numerical analysis, differential equations |
| Non-uniform rational basis spline (NURBS) | Varies | CAD,Precise modeling of curved shapes,animations |
| Open Uniform B-spline | Varies | Path design in robotics, CAD for exact endpoints |
| Closed B-spline Curve | Varies | Animation loops, gear design, closed-path modeling |

Different types of B-splines

# Differences between Curve fitting and Surface fitting

| **Aspect** | **Curve Fitting** | **Surface Fitting** |
| --- | --- | --- |
| Dimensionality | 2D(one independent variable x and one dependent variable y) | 3D(twoindependent variables,typically x and y, and one dependent variable z) |
| Objective | Finds a smooth curve that best fits a set of data points along a single line or path. | Finds a smooth surface that best fits a set of scattered points in a 3D space. |
| Result | Produces a line or curve | Produces a 3D shape or surface |
| Applications | Trend lines in data plots,Signal processing in 1D,Time-series analysis | Topographical maps from elevation data,3D reconstruction in CAD, medical imaging |

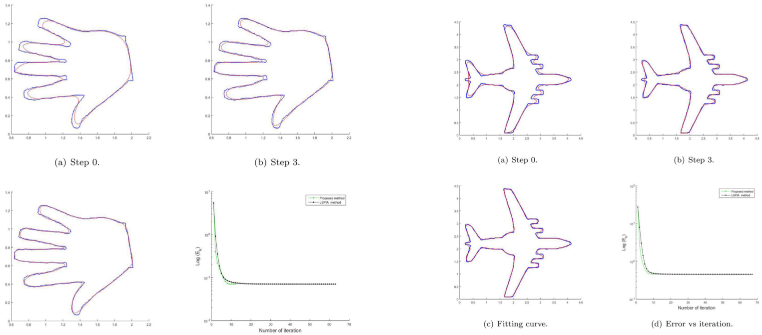
# Literature Review 1

- [1] *Ebrahimi* [1] used a composite iterative procedure for progressive and iterative approximation. This method generated a sequence of matrices based on the **Schulz iterative method**, which was used in adjusting vectors. A comparative study of experimental results highlighted various advantages of the proposed technique, such as a faster convergence rate and low approximation errors. To reduce time complexity, this model computed $Z^{k+1} = Z^k(2I - AZ^k)$in each iteration.
- [2] *Svajunas* [4] compared several **HPLSPIA** methods for matrix inverse approximation and showed that polynomial factorizations used in the hyperpower iterative methods significantly affected the efficiency of the HPLSPIA methods. This method was also applied to discuss the complexity of tensor product B-splines.

[1] *Ebrahimi and Loghmani. A composite iterative procedure with fast convergence rate for the progressive-iteration approximation of curves.J.Comput.Appl.Math.359(2019),1-15.*

[2] Svajunas sajavicius. Hyperpower last squares progressive iterative approximation.J.Comput.Appl.Math.422(2023),114888.

# Figure




Fitting curves[3]

[1]

[3] Ebrahimi and Loghmani.A composite iterative procedure with fast convergence rate for the progressive-iteration approximation of curves.J.Comput.Appl.Math.359(2019),1-15.

# Iterative methods for the matrix inverse approximation

| Method | Iterative Scheme[5] | r |
|---|---|---|
| HP2 | $$R^k = I - (A^T A).W^k$$ $$W^{(k+1)} = W^{(k)}(I + R^{(k)})$$ | 2 |
| HP3 | $$R^{(k)} = I - (A^T A)W^{(k)}$$ $$W^{(k+1)} = W^{(k)}\left(I + R^{(k)}\left(I + R^{(k)}\right)\right)$$ | 3 |
| HP5 | $$R^{(k)} = I - (A^T A)W^{(k)}$$ $$W^{(k+1)} = W^{(k)}(I + R^{(k)}(I + R^{(k)}(I + R^{(k)}(I + R^{(k)}))))$$ | 5 |

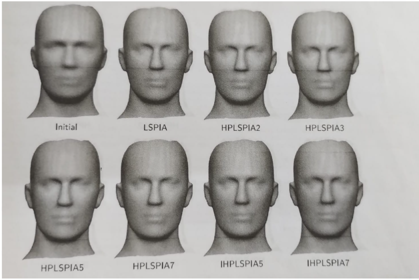
| Method | Iterative Scheme | r |
|---|---|---|
| HP7 | $R^{(k)} = I - (A^T A)W^{(k)}$ <br> $W^{(k+1)} = W^{(k)}(I + R^{(k)}(I + R^{(k)}(I + R^{(k)}(I + R^{(k)}(I + R^{(k)}(I + R^{(k)}))))))$ | 7 |
| [4]IHP5[5] | $R^{(k)} = I - (A^T A)W^{(k)}$ <br> $S^{(k)} = R^{(k)}R^{(k)}$ <br> $M^{(k)} = I + R^{(k)} + S^{(k)}(I + R^{(k)} + S^{(k)})$ <br> $W^{(k+1)} = W^{(k)}M^{(k)}$ | 5 |

[4] Marko D.and Miodrag S. Hyper-power method for the computation of outer inverse.J.Comput.Appl.Math.278(2015),110-118.

| Method | Iterative Scheme | r |
| --- | --- | --- |
| [5]IHP7[6] | $$\begin{aligned} R^{(k)} &= I - (A^T A)W^{(k)} \\ S^{(k)} &= R^{(k)}R^{(k)} \\ T^{(k)} &= (S^{(k)} + R^{(k)} + I)(S^{(k)} + R^{(k)}) \\ M^{(k)} &= (T^{(k)} - S^{(k)} - R^{(k)} - R^{(k)} \\ &\quad -2I)(T^{(k)} - 8S^{(k)} + 4R^{(k)} - 3I) \\ &\quad -6S^{(k)} + 9R^{(k)} - 1 \\ W^{(k+1)} &= W^{(k)}M^{(k)} \end{aligned}$$ | 7 |

**where r is order of convergent**

[5]Marko D. Generalized Schultz iterative method for the computation of outer inverse.Comput.Math.Appl67(10)(2014),1837-1847




| Method | Fitting error | Elapsed time | Iterations | Matrix-by-matrix multiplications |
|---|---|---|---|---|
| *Initial fitting surface* | | | | |
| | 9.64993E−02 | | | |
| *Final fitting surfaces* | | | | |
| LSPIA | 6.72666E−03 | 10.19 | 510 | 510 |
| HPLSPIA2 | 6.70050E−03 | 7.18 | 19 | 76 |
| HPLSPIA3 | 6.70050E−03 | 7.28 | 13 | 65 |
| HPLSPIA5 | 6.70050E−03 | 8.69 | 10 | 70 |
| HPLSPIA7 | 6.70050E−03 | 10.53 | 9 | 81 |
| IHPLSPIA5 | 6.70050E−03 | 7.20 | 10 | 60 |
| IHPLSPIA7 | 6.70050E−03 | 5.95 | 7 | 49 |

The initial fitting and the final fitting surfaces obtained using the standard LSPIA and different HPLSPIA method.

# Literature Review 2

- [6]*Lizheng Lu*[2]presented a new and efficient method for **weight progressive iteration** approximations of data points using normalized positive bases. Compared to the usual progressive iteration approximation, this method had a **faster convergence rate**, achieved by choosing an optimal value for the weight. These results were also valid for tensor product surfaces.
- [7]*L*i Zhang [3] introduced a new least-squares geometric iterative fitting method for generalized B-splines and introduced two distinct types of weights. This fitting method generates a series of curves by iteratively adjusting control points. He implemented internal and external weights to address various requirements in practical applications. As a result, this fitting method provided greater flexibility and could efficiently manage large-scale datasets.

[6]Lizheng Lu. Weighted progressive iteration approximation and convergence analysis. CADG27(2)(2010),129-137.

[7]Li Zhang and Xianyu Ge, Jieqing Tan. Least square geometric iterative fitting method for generalized B-spline curves with two different kinds of weight. The Visual Computer 32(2015),1109-1120.

# Literature Review 3

- [8] *Zhenmin Yao*[9]To achieve a reduction in CPU execution time, a distributed least-squares progressive iterative approximation **(DLSPIA)** method was presented. This method involves dividing the collocation matrix into blocks, which are referred to as processors. The algorithm converges within a finite number of iterations, resulting in a decrease in CPU execution time.
- [9]*Qianqian Hu*[10] combined the iterative method of the M–P generalized inverse with the LSPIA iterative format to create a family of accelerated LSPIA iterative procedures for blending curves. The proposed method had a faster average and asymptotic convergence rates than previous methods.

[8]Zhenmin Yao and Qianqian Hu.Distributed least-squares progressive iterative approximation for blending curves and surfaces.CAD175(2024),103749.

[9]Qianqian Hu and Zhifang Wang, Zhenmin Yao, Wenqing Dong.A family of hybrid iterative approximation methods for fitting blending curves.The Visual Computer40(2023),4287-4301.

# Future Directions I

1. **Iterative Schemes for Surface Fitting**
   In the survey above, we observed the application of Least-Squares Polynomial Approximation (LSPIA) for curve fitting. Inspired by these findings, we will extend this methodology to the task of **surface fitting**.
2. **Apply Various Iterative Schemes**
   will explore various iterative optimization methods to enhance curve and surface approximations, ensuring better convergence and minimizing error.
3. **Explore Different Types of Splines**
   In addition to polynomial approximation, we will also explore various spline-based approximations, such as B-splines, cubic splines, and NURBS. By selecting different types of splines, our goal is to create a smoother and more flexible surface that better accommodates complex data structures.

# Future Directions II

4. **Higher-Order Iterative Schemes**
   We aim to develop advanced higher-order iterative schemes that prioritize fast convergence (CGT) and decrease the computational time needed to reach a solution.
5. **Distributed LSPIA**
   We implement this method for Hyper Power LSPIA to enhance the blending of curves and surfaces. This advanced approach incorporates function values and higher-order derivatives into the approximation process. The benefits of using Hyper Power LSPIA include **increased smoothness** and **improved control over curvature**.

# References I


Ebrahimi and Loghmani.A composite iterative procedure with a fast convergence rate for the progressive-iteration approximation of curves. *J.Comput.Appl.Math.*359(2019),1-15.

Lizheng Lu. Weighted progressive iteration approximation and convergence analysis. *CADG*27(2)(2010),129-137.

Li Zhang and Xianyu Ge, Jieqing Tan . Least square geometric iterative fitting method for generalized B-spline curves with two different kinds of weight. *The Visual Computer* 32(2015),1109-1120.

Svajunas sajavicius. Hyperpower last squares progressive iterative approximation.*J.Comput.Appl.Math.* 422(2023),114888.

Marko D and Miodrag S. Hyper-power method for the computation of outer inverse.*J.Comput.Appl.Math.* 278(2015),110-118.

# References II


- Marko D. Generalized Schultz iterative method for the computation of outer inverse.*Comput. Math. Appl*67(10)(2014),1837-1847
- Hearn,Baker, M. P. (2010). Computer Graphics with OpenGL (3rd ed.).
- Montgomery, D. C., Peck, E. A., Vining, G. G. (2012). Introduction to Linear Regression Analysis (5th ed.).
- Zhenmin Yao and Qianqian Hu.Distributed least-squares progressive iterative approximation for blending curves and surfaces.*CAD*175(2024),103749.
- Qianqian Hu and Zhifang Wang, Zhenmin Yao, Wenqing Dong.A family of hybrid iterative approximation methods for fitting blending curves*The Visual Computer*40(2023),4287-4301.

THANK YOU!