Gauss-Seidel 最小二乘渐进迭代逼近

Yusuf Fatihu Hamza¹⁾, 蒋旖旎¹⁾, 蔺宏伟^{1,2)*}

¹⁾ (浙江大学数学科学学院 杭州 310027)

²⁾ (浙江大学 CAD&CG 国家重点实验室 杭州 310058)
(hwlin@zju.edu.cn)

摘要: 几何迭代法, 即渐进迭代逼近(progressive-iterative approximation, PIA), 作为一种有效的数据拟合方法, 吸引了众多研究者的关注, 并获得广泛的应用. 针对经典 LSPIA 算法收敛速度较慢的问题, 提出一种基于 Gauss-Seidel 迭代方法的快速 PIA 算法, 称为 GS-LSPIA. 首先, 从给定的数据点中选取拟合曲线的控制点; 然后, 采用累加弦长法参数化给定数据点; 最后, GS-LSPIA 通过迭代地调整控制点来生成一系列拟合曲线(曲面), 并且保证了生成的曲线(曲面)的极限是对于给定数据点的最小二乘拟合结果. 在多个曲线曲面拟合上的实验结果表明, 为达到相同的拟合精度, GS-LSPIA 算法比 LSPIA 算法需要更少的步骤和更短的运算时间. 因此, GS-LSPIA 是有效的, 而且具有比 LSPIA 算法更快的收敛速度.

关键词: 渐进迭代逼近; Gauss-Seidel 迭代法; 数据拟合

中图分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2021.18289

Gauss-Seidel Progressive and Iterative Approximation for Least Squares Fitting

Yusuf Fatihu Hamza¹⁾, Jiang Yini¹⁾, and Lin Hongwei^{1,2)*}

¹⁾ (School of Mathematical Sciences, Zhejiang University, Hangzhou 310027)

²⁾ (State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310058)

Abstract: Progressive-iterative approximation (PIA) is an efficient method for data fitting that attracts the attention of many researchers and has a wide range of applications. However, the convergence rate of LSPIA is prolonged. In this study, we design a fast PIA format based on the Gauss-Seidel iterative method named Gauss-Seidel progressive and iterative approximation for least squares curve and surface fitting (GS-LSPIA). Firstly, the control points of the fitting curve (surface) are selected from the given data points. Then, the chord length method is used to assign the parameters of the given data points. GS-LSPIA generates a series of fitting curves (surfaces) by refining the control points iteratively, and the limit of the generated curve (surface) is the least square fitting result to the given data points. Several experimental results presented in this paper demonstrate that, to achieve the same accuracy for GS-LSPIA and LSPIA, GS-LSPIA required fewer steps and shorter running time compared with LSPIA. Thus, the proposed GS-LSPIA is efficient and has a faster convergence rate compared with the LSPIA algorithm.

Key words: progressive-iterative approximation; Gauss-Seidel iterative method; data fitting

曲线和曲面的拟合问题出现在许多领域, 如计算机图形学、计算机辅助设计和图像处理等. 与

传统求解线性方程组的方法相比, 渐进迭代逼近 (progressive-iterative approximation, PIA) 避免了直

收稿日期: 2020-03-28; 修回日期: 2020-05-04. 基金项目: 国家自然科学基金(61872316, 61932018). Yusuf Fatihu Hamza(1986—), 男, 博士研究生, 主要研究方向为计算机辅助几何设计和图像处理; 蒋旖旎(1996—), 女, 博士研究生, 主要研究方向为计算机图形学和深度学习; 蔺宏伟(1973—), 男, 博士, 教授, 博士生导师, CCF 会员, 论文通讯作者, 主要研究方向为几何设计.

接求解线性方程组的诸多缺点,如计算成本高. PIA 算法会生成一系列逼近曲线(曲面),并且在每次迭代后对曲线(曲面)进行调整,直到获得拟合给定数据点的曲线(曲面). PIA 算法虽具有直观的几何含义且易于编程,但其收敛速度较慢. 本文提出一种基于 Gauss-Seidel 迭代方法的、快速高效的 PIA 算法,称为 Gauss-Seidel 最小二乘几何迭代法(Gauss-Seidel progressive and iterative approximation for least square B-spline curve and surface fitting, GS-LSPIA). 实验结果表明,GS-LSPIA 算法的收敛速度快于经典的 LSPIA 算法.

齐东旭等^[1]于 1975 年发现了均匀三次 B 样条曲线的盈亏修正性质; de Boor^[2]于 1979 年也独立发现该性质. 2004 年, Lin 等^[3]证明非均匀 B 样条曲线曲面也满足该性质. 随后, Lin 等^[4]证明了具有规范化全正基(normalized totally positive, NTP)的曲线曲面都具有该性质,并将其命名为 PIA. Delgado 等^[5]指出,规范化 B 样条基在 NTP 中具有最快的收敛速度. 史利民等^[6]证明了 NURBS 同样具有此性质. Lin^[7]通过调整混合曲线(曲面)的一部分控制点,提出了局部 PIA 算法,算法生成的极限曲线(曲面)插值数据点的一部分.

近年来,不断有学者对 PIA 算法进行改进. 在经典的 PIA 中,控制点个数等于数据点个数. Lin 等^[8]设计了一种扩展的渐进迭代逼近(extended progressive and iterative approximation, EPIA)算法来拟合大规模数据点,它允许控制点个数小于数据点个数. 类似于 EPIA, Deng 等^[9]提出的最小二乘几何迭代法(progressive and iterative approximation for least squares B-spline curve and surface fitting, LSPIA)是一种适合大规模数据的有效算法,它的极限曲线曲面是对给定数据点的最小二乘拟合结果. Lin 等^[10]证明了 LSPIA 算法可以求解奇异的最小二乘拟合方程组. Lu^[11]提出了加权 PIA 算法来加快经典 PIA 的收敛速度. 基于广义 B 样条,另一种具有不同权重的 LSPIA 算法被提出以拟合更复杂的数据点^[12-13]. 此外, Liu 等^[14]提出了用于正则化最小二乘双变量 B 样条曲面拟合算法(progressive iterative approximation for regularized least square bivariate B-spline surface fitting, RLSPIA),将单变量 NTP 的 PIA 性质推广至线性相关的非张量积型二元 B 样条基. Ebrahimi 等^[15]提出了一种复合型 PIA 算法,采用 Schulz 迭代方法计算调整向量.

PIA 算法还被扩展至细分格式,如 Catmull-Clark, Loop 和 Doo-Sabin 细分曲面,称为渐进插值

(progressive interpolation, PI). Chen 等^[16]基于 Catmull-Clark 细分设计了渐进插值逼近; Cheng 等^[17]提出了 Loop 细分曲面的 PI 算法; Deng 等^[18]给出了 Loop 细分曲面的加权 PIA,以加快 PI 算法在 Loop 细分曲面上的收敛速度; Wang 等^[19]提出了用于细分曲面插值的 Gauss-Seidel 几何迭代法(Gauss-Seidel progressive iterative approximation, GS-PIA); 刘晓艳等^[20]给出了非均匀三次 B 样条曲线插值的 GS-PIA 算法. 最近, PIA 算法被 Zhang 等^[21]应用于单帧图像超分辨率重建; Hamza 等^[22]用于隐式曲线和曲面重构.

1 GS-LSPIA 及其收敛性

1.1 GS-LSPIA 曲线拟合算法

给定一个有序点列 $\mathbf{Q} = \{\mathbf{q}_i, i = 0, 1, \dots, m\}$, 为每一个点 $\mathbf{q}_i$ 分配一个参数值 $\{t_i, i = 0, 1, \dots, m\}$, 满足

$$0 = t_0 < t_1 < \dots < t_m = 1.$$

令 $\mathbf{P}^0 = \{\mathbf{p}_i^0, i = 0, 1, \dots, n\} (m \geq n)$ 为初始控制点列, X^0 为初始控制多边形, 构造一条初始混合曲线 $\mathbf{P}^0(t) = \sum_{i=0}^n B_i(t) \mathbf{p}_i^0, t \in [0, 1]$. 其中, $\{B_i(t), i = 0, 1, \dots, n\}$ 是一族基, 这族基在参数序列 $\{0 = t_0 < t_1 < \dots < t_m = 1\}$ 上的配置矩阵为

$$\mathbf{B} = \begin{bmatrix} B_0(t_0) & B_1(t_0) & \dots & B_n(t_0) \\ B_0(t_1) & B_1(t_1) & \dots & B_n(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ B_0(t_m) & B_1(t_m) & \dots & B_n(t_m) \end{bmatrix} \quad (1)$$

在 GS-LSPIA 的第 1 次迭代中, 首先计算对应于数据点的差向量 $\delta_j^0 = \mathbf{q}_j - \sum_{h=0}^n B_h(t_j) \mathbf{p}_h^0, j \in I_0$. 其中, I_0 为 $B_0(t_l) \neq 0$ 的下标 l 的集合. 然后构造对应于控制点的差向量

$$\mathbf{d}_0^0 = \frac{\sum_{j \in I_0} B_0(t_j) \delta_j^0}{\sum_{h \in I_0} B_0(t_h) B_0(t_h)};$$

则新的控制点为 $\mathbf{p}_0^1 = \mathbf{p}_0^0 + \mathbf{d}_0^0$.

类似地, 设 $\mathbf{P}_i^0 = \{\mathbf{p}_0^1, \dots, \mathbf{p}_{i-1}^1, \mathbf{p}_i^0, \dots, \mathbf{p}_n^0\}_{i=0}^n$ 为控制多边形 X_i^0 的控制点. 为计算第 i 个控制点, 首先计算对应于数据点的差向量

$$\delta_j^0 = \mathbf{q}_j - \left(\sum_{h=0}^{i-1} B_h(t_j) \mathbf{p}_h^1 + \sum_{h=i}^n B_h(t_j) \mathbf{p}_h^0 \right), j \in I_i.$$

其中, I_l 为 $B_l(t_l) \neq 0$ 的下标 l 的集合. 然后, 构造对应于控制点的差向量

$$d_i^0 = \frac{\sum_{j \in I_l} B_i(t_j) \delta_j^0}{\sum_{h \in I_l} B_i(t_h) B_i(t_h)};$$

则新的控制点为 $p_i^1 = p_i^0 + d_i^0$.

当 $i = n$ 时, 第 1 次迭代完成. 假设控制多边形 X^1 的控制顶点为 $P^1 = \{p_i^1\}_{i=0}^n$, 其中, $p_i^1 = p_i^0 + d_i^0$, $i = 0, 1, \dots, n$; 得到新的迭代曲线为

$$P^1(t) = \sum_{i=0}^n B_i(t) p_i^1, t \in [t_0, t_m].$$

现假设在 k 次迭代后生成的第 k 次曲线为 $P^k(t)$, 其控制多边形 X^k 的控制顶点为 $P^k = \{p_i^k\}_{i=0}^n$. 首先计算对应于数据点的差向量

$$\delta_j^k = q_j - \sum_{h=0}^n B_h(t_j) p_h^k, j \in I_0.$$

其中, I_0 为 $B_0(t_l) \neq 0$ 的下标 l 的集合. 然后构造对应于控制点的差向量

$$d_0^k = \frac{\sum_{j \in I_0} B_0(t_j) \delta_j^k}{\sum_{h \in I_0} B_0(t_h) B_0(t_h)};$$

则新的控制点为 $p_0^{k+1} = p_0^k + d_0^k$.

类似地, 设 $P_i^k = \{p_0^{k+1}, \dots, p_{i-1}^{k+1}, p_i^k, \dots, p_n^k\}_{i=0}^n$ 为控制多边形 X_i^k 的控制点. 为计算第 i 个控制点, 首先计算对应于数据点的差向量

$$\delta_j^k = q_j - \left(\sum_{h=0}^{i-1} B_h(t_j) p_h^{k+1} + \sum_{h=i}^n B_h(t_j) p_h^k \right), j \in I_i.$$

其中, I_i 为 $B_i(t_l) \neq 0$ 的下标 l 的集合. 然后, 构造对应于控制点的差向量

$$d_i^k = \frac{\sum_{j \in I_i} B_i(t_j) \delta_j^k}{\sum_{h \in I_i} B_i(t_h) B_i(t_h)};$$

则新的控制点为 $p_i^{k+1} = p_i^k + d_i^k$.

当 $i = n$ 时, 第 $k+1$ 次迭代完成. 假设控制多边形 X^{k+1} 的控制顶点为 $P^{k+1} = \{p_i^{k+1}\}_{i=0}^n$, 其中,

$$p_i^{k+1} = p_i^k + d_i^k, i = 0, 1, \dots, n \quad (2)$$

得到新的迭代曲线为

$$P^{k+1}(t) = \sum_{i=0}^n B_i(t) p_i^{k+1}, t \in [t_0, t_m].$$

图 1 所示为 GS-LSPIA 进行曲线拟合的迭代过程. 其中, 红色点为原始数据点, 蓝色点为调整的控制点. 当迭代次数趋于无穷时, 拟合曲线逼近给定的数据点.

图 1 曲线拟合的 GS-LSPIA 迭代法

算法 1. GS-LSPIA 曲线拟合算法.

输入. 有序数据点集 $\mathbf{Q} = \{\mathbf{q}_j, j = 0, 1, \dots, m\}$, 拟合精度 $\varepsilon > 0$.

输出. 拟合曲线 $\mathbf{P}^k(t)$.

Step1. 对数据点进行参数化.

Step2. 选择初始控制点.

Step3. 计算配置矩阵 $\mathbf{B}$ 及矩阵 $\mathbf{A} = \mathbf{B}^T \mathbf{B}$.

Step4. 令 $k = 0$.

Step5. 令 $i = 0$.

Step6. 计算差向量

$$\delta_j^k = \mathbf{q}_j - \left(\sum_{h=0}^{i-1} B_h(t_j) \mathbf{p}_h^{k+1} + \sum_{h=i}^n B_h(t_j) \mathbf{p}_h^k \right), j \in I_i.$$

Step7. 计算调整向量 $\mathbf{d}_i^k = \frac{\sum_{j \in I_i} B_i(t_j) \delta_j^k}{\sum_{h \in I_i} B_i(t_h) B_i(t_h)}$.

Step8. 计算第 $k+1$ 次迭代曲线的新控制点 $\mathbf{p}_i^{k+1} = \mathbf{p}_i^k + \mathbf{d}_i^k$.

Step9. 若 $i = n$, 转 Step10; 否则, 令 $i = i+1$, 转 Step6.

Step10. 生成第 $k+1$ 次曲线 $\mathbf{P}^{k+1}(t) = \sum_{i=0}^n B_i(t) \mathbf{p}_i^{k+1}$.

Step11. 计算拟合误差 E_k , 若达到拟合精度, 则停止迭代; 否则, 令 $k = k+1$, 转 Step5.

收敛性证明. 类似于文献[23], 可以证明上述 GS-LSPIA 算法的收敛性.

定理 1. 当基函数 $\{B_i(t), i = 0, 1, \dots, n\}$ 在参数列 $\{0 = t_0 < t_1 < \dots < t_m = 1\}$ 上的配置矩阵 $\mathbf{B}$ (1) 是非奇异的, 上述 GS-LSPIA 迭代算法是收敛的, 并且极限曲线 $\mathbf{P}^\infty(t)$ 是给定数据点 $\mathbf{Q} = \{\mathbf{q}_i, i = 0, 1, \dots, m\}$ 的最小二乘逼近.

证明. 由上述迭代格式, 得到曲线序列 $\{\mathbf{P}^k(t), k = 0, 1, \dots\}$. 将式(2)展开, 得

$$\begin{aligned} \mathbf{p}_i^{k+1} &= \mathbf{p}_i^k + \mathbf{d}_i^k = \mathbf{p}_i^k + \frac{\sum_{j=0}^n B_i(t_j) \delta_j^k}{\sum_{h=0}^n B_i(t_h) B_i(t_h)} = \\ &= \mathbf{p}_i^k + \frac{\sum_{j=0}^n B_i(t_j)}{\sum_{h=0}^n B_i(t_h) B_i(t_h)} \left[\mathbf{q}_j - \left(\sum_{h=0}^{i-1} B_h(t_j) \mathbf{p}_h^{k+1} + \sum_{h=i}^n B_h(t_j) \mathbf{p}_h^k \right) \right] = \end{aligned}$$

$$\frac{1}{\sum_{h=0}^n B_i(t_h) B_i(t_h)} \left[\sum_{j=0}^n B_i(t_j) \mathbf{q}_j - \sum_{j=0}^n B_i(t_j) \left(\sum_{h=0}^{i-1} B_h(t_j) \mathbf{p}_h^{k+1} + \sum_{h=i+1}^n B_h(t_j) \mathbf{p}_h^k \right) \right].$$

于是, GS-LSPIA 曲线拟合算法可改写为

$$\mathbf{P}^{k+1} = \mathbf{D}^{-1} (\mathbf{B}^T \mathbf{Q} - \mathbf{L} \mathbf{P}^{k+1} - \mathbf{U} \mathbf{P}^k) = -(\mathbf{L} + \mathbf{D})^{-1} \mathbf{U} \mathbf{P}^k + (\mathbf{L} + \mathbf{D})^{-1} \mathbf{B}^T \mathbf{Q} \quad (3)$$

其中, $\mathbf{B}$ 为配置矩阵式(1); $\mathbf{D}$, $\mathbf{L}$ 和 $\mathbf{U}$ 是从矩阵 $\mathbf{A} = \mathbf{B}^T \mathbf{B}$ 分解出来的矩阵, 即 $\mathbf{A} = \mathbf{B}^T \mathbf{B} = \mathbf{D} + \mathbf{L} + \mathbf{U}$. 其中, $\mathbf{D}$ 是对角阵; $\mathbf{L}$ 和 $\mathbf{U}$ 分别是严格下三角矩阵和严格上三角矩阵. 根据定理条件, 配置矩阵 $\mathbf{B}$ 是非奇异的, 因此矩阵 $\mathbf{A} = \mathbf{B}^T \mathbf{B}$ 是对称正定阵.

为了说明迭代式(3)的收敛性, 将方程两边同时减去 $\mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q}$, 得

$$\begin{aligned} \mathbf{P}^{k+1} - \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q} &= \\ &= -(\mathbf{L} + \mathbf{D})^{-1} \mathbf{U} \mathbf{P}^k + (\mathbf{L} + \mathbf{D})^{-1} \mathbf{A} \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q} - \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q} = \\ &= [-(\mathbf{L} + \mathbf{D})^{-1} \mathbf{U}] (\mathbf{P}^k - \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q}) = \\ &= [-(\mathbf{L} + \mathbf{D})^{-1} \mathbf{U}]^2 (\mathbf{P}^{k-1} - \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q}) = \dots = \\ &= \mathbf{G}^{k+1} (\mathbf{P}^0 - \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q}) \end{aligned} \quad (4)$$

这里 $\mathbf{G} = [-(\mathbf{L} + \mathbf{D})^{-1} \mathbf{U}]$. 如果 $\rho(\mathbf{G}) < 1$, 则迭代方程式(4)收敛, 其中, $\rho(\mathbf{G})$ 为 $\mathbf{G}$ 的谱半径.

由于 $\mathbf{D}$ 是 Hermite 正定的, 则

$$\mathbf{G}_1 = \mathbf{D}^{\frac{1}{2}} \mathbf{G} \mathbf{D}^{-\frac{1}{2}} = -(\mathbf{I} + \mathbf{L}_1)^{-1} \mathbf{U}_1.$$

其中, $\mathbf{L}_1 = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}}$; $\mathbf{U}_1 = \mathbf{D}^{-\frac{1}{2}} \mathbf{U} \mathbf{D}^{-\frac{1}{2}}$. 因为 $\mathbf{G}$ 与 $\mathbf{G}_1$ 特征值相同, 只需证明 $\rho(\mathbf{G}_1) < 1$. 令 $\mathbf{x}$ 为 $\mathbf{G}_1$ 的特征值 λ 所对应的特征向量(其中, $\mathbf{x}^H \mathbf{x} = 1$, $\mathbf{x}^H$ 为 $\mathbf{x}$ 的共轭转置), 即

$$\begin{aligned} -(\mathbf{I} + \mathbf{L}_1)^{-1} \mathbf{U}_1 \mathbf{x} &= \lambda \mathbf{x}, \\ \mathbf{U}_1 \mathbf{x} &= -\lambda (\mathbf{I} + \mathbf{L}_1) \mathbf{x}. \end{aligned}$$

等式两边同时左乘 $\mathbf{x}^H$, 得到

$$\begin{aligned} \mathbf{x}^H \mathbf{U}_1 \mathbf{x} &= -\lambda \mathbf{x}^H (\mathbf{I} + \mathbf{L}_1) \mathbf{x}, \\ \lambda &= \frac{-\mathbf{x}^H \mathbf{U}_1 \mathbf{x}}{\mathbf{x}^H (\mathbf{I} + \mathbf{L}_1) \mathbf{x}} \end{aligned} \quad (5)$$

设 $\mathbf{x}^H \mathbf{L}_1 \mathbf{x} = a + bi$, 则

$$\mathbf{x}^H \mathbf{U}_1 \mathbf{x} = \mathbf{x}^H \mathbf{L}_1^H \mathbf{x} = (\mathbf{x}^H \mathbf{L}_1 \mathbf{x})^H = \overline{\mathbf{x}^H \mathbf{L}_1 \mathbf{x}} = a - bi.$$

由于 $\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} = \mathbf{I} + \mathbf{L}_1 + \mathbf{U}_1$ 是 Hermite 正定的, 则 $0 < 1 + \mathbf{x}^H \mathbf{L}_1 \mathbf{x} + \mathbf{x}^H \mathbf{U}_1 \mathbf{x} = 1 + 2a$. 于是, 通过式(5)有

$$\lambda = \frac{-a + bi}{1 + a + bi},$$

$$\|\lambda\|^2 = \lambda \bar{\lambda} = \frac{a^2 + b^2}{1 + 2a + a^2 + b^2} < 1.$$

这意味着 $\rho(\mathbf{G}) = \rho(\mathbf{G}_1) < 1$. 因此 $\lim_{k \rightarrow \infty} \mathbf{G}^k = 0$.

由式(4)可知,

$$\mathbf{P}^\infty = \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q} + \mathbf{G}^\infty (\mathbf{P}^0 - \mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q}) =$$

$$\mathbf{A}^{-1} \mathbf{B}^T \mathbf{Q} = (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \mathbf{Q}, \quad (\mathbf{A} = \mathbf{B}^T \mathbf{B}).$$

这等价于 $\mathbf{B}^T \mathbf{B} \mathbf{P}^\infty = \mathbf{B}^T \mathbf{Q}$. 于是, GS-LSPIA 迭代格式收敛. 极限曲线是初始数据点的最小二乘拟合结果, 定理得证. 证毕.

1.2 GS-LSPIA 曲面拟合算法

下面将 GS-LSPIA 算法推广至混合曲面.

给定点集 $\mathbf{Q} = \{\mathbf{q}_i\}_{i=0}^m$, 将每个数据点 $\mathbf{q}_i$ 对应的参数值记为 $\{(u_i, v_i)\}_{i=0}^m$. 需要指出的是, 如果给定的是有序点集 $\mathbf{Q} = \{\mathbf{q}_{ij}\}_{i=0, j=0}^{m_1, m_2}$, 则可以把这个有序点集按照字典序排列成一维点集 $\mathbf{Q} = \{\mathbf{q}_i\}_{i=0}^m$. 假设 $\mathbf{P}^0 = \{\mathbf{p}_{hl}^0\}_{h=0, l=0}^{n_u, n_v}$ 为初始控制点集, X^0 为初始控制多边形, 并构造初始混合曲面 $\mathbf{P}^0(u, v) = \sum_{h=0}^m \sum_{l=0}^n B_h(u) B_l(v) \mathbf{p}_{hl}^0$. 将上述基函数和控制点按照字典序排成一维数组, 并重新标记

$$[b_0(u, v), b_1(u, v), \dots, b_{(n_u+1)(n_v+1)}(u, v)] =$$

$$[B_0(u)B_0(v), B_0(u)B_1(v), \dots, B_{n_u}(u)B_{n_v}(v)],$$

$$[\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_{(n_u+1)(n_v+1)}] = [\mathbf{p}_{00}, \mathbf{p}_{01}, \dots, \mathbf{p}_{n_u, n_v}];$$

则初始混合曲面可改写为

$$\mathbf{P}^0(u, v) = \sum_{i=0}^{(n_u+1)(n_v+1)} b_i(u, v) \mathbf{p}_i^0 \quad (6)$$

此时, 基函数的配置矩阵可以表示为

$$\mathbf{C} = \begin{bmatrix} b_0(u_0, v_0) & b_1(u_0, v_0) & \cdots & b_{(n_u+1)(n_v+1)}(u_0, v_0) \\ b_0(u_1, v_1) & b_1(u_1, v_1) & \cdots & b_{(n_u+1)(n_v+1)}(u_1, v_1) \\ \vdots & \vdots & \vdots & \vdots \\ b_0(u_m, v_m) & b_1(u_m, v_m) & \cdots & b_{(n_u+1)(n_v+1)}(u_m, v_m) \end{bmatrix} \quad (7)$$

从初始曲面式(6)开始, 张量积曲面的 GS-LSPIA 迭代格式与曲线的 GS-LSPIA 完全一样.

经过 GS-LSPIA 迭代过程, 可以得到一个曲面序列 $\{\mathbf{P}(u, v), k = 0, 1, \dots\}$. 类似于曲线情况, 可以

证明, 上述曲面序列收敛, 且收敛到对给定点集 $\mathbf{Q} = \{\mathbf{q}_i\}_{i=0}^m$ 的最小二乘拟合曲面.

定理 2. 当配置矩阵 $\mathbf{C}$ 是非奇异的, 上述张量积曲面的 GS-LSPIA 迭代算法是收敛的, 并且极限曲面 $\mathbf{P}^\infty(u, v)$ 是给定数据点集的最小二乘逼近.

上述定理的证明与定理 1 类似.

2 算法实现与数值实例

2.1 算法实现

在本文的实验中, 以 B 样条曲线曲面为例来验证 GS-LSPIA 算法.

对于曲线情形, 类似于文献[9], 选取初始控制点 $\mathbf{P} = \{\mathbf{p}_j, j = 0, 1, \dots, n\}$ 和节点向量 $\{0, 0, 0, 0, u_4, u_5, \dots, u_n, 1, 1, 1, 1\}$. 给定一个有序点集 $\mathbf{Q} = \{\mathbf{q}_i, i = 0, 1, \dots, m\}$, 采用累加弦长法分配对应参数 $\{t_i, i = 0, 1, \dots, m\}$ [24], 即

$$\begin{cases} t_0 = 0, & t_m = 1 \\ t_i = t_{i-1} + \frac{\|\mathbf{q}_i - \mathbf{q}_{i-1}\|}{L}, & i = 1, 2, \dots, m-1 \end{cases}$$

其中, $L = \sum_{i=0}^m \|\mathbf{q}_i - \mathbf{q}_{i-1}\|$ 为总弦长. 定义三次 B 样条曲线 $\mathbf{P}(t) = \sum_{i=0}^n \mathbf{p}_i B_{i,3}(t)$, 其中, $B_{i,3}$ 是三次 B 样条基函数, 其定义在节点向量 $\{0, 0, 0, 0, u_4, u_5, \dots, u_n, 1, 1, 1, 1\}$ 上. 其中,

$$u_{j+3} = (1 - \alpha)t_{i-1} + \alpha t_i, \quad j = 1, \dots, n-3,$$

$$i = [jd], \quad \alpha = jd - i, \quad d = \frac{m+1}{n-2}.$$

其中, $[jd]$ 为不大于 jd 的最大整数. 此外, 选择初始控制点 $\mathbf{P} = \{\mathbf{p}_i, i = 0, 1, \dots, n\}$ 如

$$\begin{cases} \mathbf{p}_0 = \mathbf{q}_0 \\ \mathbf{p}_n = \mathbf{q}_m \\ \mathbf{p}_i = \mathbf{q}_{f(i)}, \quad i = 1, 2, \dots, n-1 \end{cases}$$

其中, $f(i) = \left\lceil \frac{(m+1)i}{n} \right\rceil$.

对于曲面情形, 给定有序点集 $\mathbf{Q} = \{\mathbf{q}_{ij}\}_{i=0, j=0}^{m_1, m_2}$, 每个数据点对应参数值 $\{u_i, v_j\}_{i=0, j=0}^{m_1, m_2}$, 满足

$$u_i = \frac{1}{m_2} \sum_{j=0}^{m_2} t_{ij}, \quad i = 0, 1, \dots, m_1,$$

$$v_j = \frac{1}{m_1} \sum_{i=0}^{m_1} s_{ij}, \quad j = 0, 1, \dots, m_2.$$

其中, 对于每个 $i=0,1,\dots,m_1$, 采用累加弦长法给数据点 $\{q_{ij}\}_{j=0}^{m_2}$ 分配参数 $\{t_{ij}\}_{j=0}^{m_2}$; 对于每个 $j=0,1,\dots,m_2$, 采用累加弦长法给数据点 $\{q_{ij}\}_{i=0}^{m_1}$ 分配参数 $\{s_{ij}\}_{i=0}^{m_1}$. 定义三次 B 样条曲面

$$P(u,v) = \sum_{h=0}^{n_u} \sum_{l=0}^{n_v} B_{h,3}(u) B_{l,3}(v) p_{hl}.$$

其中, $B_{h,3}$ 与 $B_{l,3}$ 是三次 B 样条基函数分别定义在节点向量 $\{0,0,0,0,\bar{u}_4,\bar{u}_5,\dots,\bar{u}_{n_u},1,1,1,1\}$ 和 $\{0,0,0,0,\bar{v}_4,\bar{v}_5,\dots,\bar{v}_{n_v},1,1,1,1\}$ 上. 其中,

$$\bar{u}_{j+3} = (1-\alpha)u_{i-1} + \alpha u_i, \quad j=1,\dots,n_u-3,$$

$$i = [jd_u], \quad \alpha = jd_u - i, \quad d_u = \frac{m_1+1}{n_u-2};$$

并且

$$\bar{v}_{j+3} = (1-\alpha)v_{i-1} + \alpha v_i, \quad j=1,\dots,n_v-3,$$

$$i = [jd_v], \quad \alpha = jd_v - i, \quad d_v = \frac{m_2+1}{n_v-2}.$$

此外, 选取初始控制点 $P = \{p_{hl}\}_{h=0,l=0}^{n_u,n_v}$ 如

$$\begin{cases} p_{0,0} = q_{0,0} \\ p_{n_u,0} = q_{m_1,0} \\ p_{0,n_v} = q_{0,m_2} \\ p_{n_u,n_v} = q_{m_1,m_2} \\ p_{i,j} = q_{f_1(i),f_2(j)}, \quad i=1,2,\dots,n_u-1, \quad j=1,2,\dots,n_v-1 \end{cases}.$$

$$\text{其中, } f_1(i) = \left\lfloor \frac{(m_1+1)i}{n_u} \right\rfloor; \quad f_2(j) = \left\lfloor \frac{(m_2+1)j}{n_v} \right\rfloor.$$

2.2 数值实例

下面通过几个平面曲线和三维曲面的例子验证 GS-LSPIA 算法的有效性. 首先, 令第 k 次迭代的拟合误差为 $E_k = \max_j \{e_j^k\}$, $e_j^k = \left\| q_j - \sum_{i=0}^n B_i(t_j) p_i^k \right\|$. 对于曲线拟合, 数据点用红色表示, 拟合曲线用蓝色表示.

例 1. 图 2 描绘了从大写字母“G”形状上采样得到的 269 个数据点, 从中选取 192 个作为控制点.

图 2 利用 GS-LSPIA 算法对 G 形数据点进行曲线拟合

例 2. 图 3a 给出数据点集 Flower, 包含 342 个数据点, 从中选取 285 个作为控制点.

例 3. 图 3d 给出数据点集 Dolphin, 包含 331 个数据点, 从中选取 276 个作为控制点.

图 3 的第 2 列从上至下依次是模型 Flower 和 Dolphin 由 GS-LSPIA 生成的拟合曲线, 第 3 列显示了初始数据点叠加的拟合曲线. 表 1 列出了实现拟合精度 E_k 依次小于 $10^{-1}, 10^{-2}, \dots, 10^{-5}$ 时 GS-LSPIA 和 LSPIA^[9] 分别所需的迭代次数 k , 以及达到精度 $E_k < 10^{-5}$ 所需的运算时间. 此外,

为了说明 GS-LSPIA 的收敛速度, 图 4 分别给出了例 1~例 3 的迭代次数对拟合误差的影响. 可以看出, GS-LSPIA 的收敛速度要快于 LSPIA (表 1 和图 4).

下面通过 3 个例子来评估 GS-LSPIA 曲面拟合的效率.

例 4. 图 5 所示是由 25×25 个数据点组成的多峰曲面模型. 从曲面上采样得到 625 个数据点, 从中选取 441 个作为控制点. 图 5a 显示了数据点叠加后的初始多峰曲面, 图 5b 和图 5c 分别描绘了本

图 3 利用 GS-LSPIA 对模型 Flower 和 Dolphin 数据点进行曲线拟合

表 1 图 2 和图 3 中数据点集在拟合精度 E_k 下的迭代次数及时间

模型	算法	迭代次数					时间/s
		$E_k < 10^{-1}$	$E_k < 10^{-2}$	$E_k < 10^{-3}$	$E_k < 10^{-4}$	$E_k < 10^{-5}$	
G-shape	LSPIA ^[9]	0	1	6	31	102	0.145 84
	GS-LSPIA	0	1	2	11	23	0.098 61
Flower	LSPIA ^[9]	0	1	5	24	87	0.166 29
	GS-LSPIA	0	1	2	5	20	0.109 44
Dolphin	LSPIA ^[9]	0	1	3	25	53	0.122 61
	GS-LSPIA	0	1	2	5	10	0.084 81

注. 时间为达到精度 $E_k < 10^{-5}$ 所需的运算时间.

图 4 不同算法的迭代次数与拟合误差关系图

图 5 利用 GS-LSPIA 拟合多峰曲面

文算法在迭代 9 次后生成的曲面和误差图.

例 5. 图 6 所示是一个参数曲面, 称为壳体曲面(shell surface), 定义为

$$\begin{cases} x = \frac{1}{5}\left(1 - \frac{v}{2\pi}\right)\cos(2v)(1 + \cos(u)) + \frac{1}{10}\cos(2v) \\ y = \frac{1}{5}\left(1 - \frac{v}{2\pi}\right)\sin(2v)(1 + \cos(u)) + \frac{1}{10}\sin(2v); \\ z = \frac{v}{2\pi} + \frac{1}{5}\left(1 - \frac{v}{2\pi}\right)\sin(u) \end{cases}$$

其中, $0 \leq u, v \leq 2\pi$. 从壳体曲面上采样得到 3 200 个数据点, 从中选取 2 226 个作为控制点. 如图 6a 所示, 采用 GS-LSPIA 算法对数据点进行拟合, 拟合曲面如图 6b 所示, 图 6c 所示为叠加给定数据点后的拟合曲面.

例 6. 图 7 所示是一个复杂的参数曲面, 称为

Tranguloid 三叶曲面, 定义为

$$\begin{cases} x = \frac{2\sin(3u)}{2 + \cos(v)} \\ y = 2\frac{\sin(u) + 2\sin(2u)}{2 + \cos\left(v + \frac{2\pi}{3}\right)}, -\pi \leq u, v \leq \pi \\ z = \frac{1}{4}(\cos(u) - 2\cos(2u))(2 + \cos(v)) \cdot \left(2 + \cos\left(v + \frac{2\pi}{3}\right)\right) \end{cases}$$

给定来自 Tranguloid 三叶曲面(tranguloid trefoil surface)上的 4 032 个数据点, 从中选取 2 809 个作为控制点. 如图 7a 所示, 本文 GS-LSPIA 算法对给定的数据点进行了准确的拟合; 图 7b 描绘了迭代 5 次后的拟合曲面; 图 7c 为叠加给定数据点后的拟合曲面.

图 6 利用 GS-LSPIA 拟合壳体曲面

图 7 利用 GS-LSPIA 拟合 Tranguloid 三叶曲面

设定拟合精度 E_k 依次小于 $10^{-1}, 10^{-2}, \dots, 10^{-5}$; 表 2 给出了 GS-LSPIA 和 LSPIA 达到给定精度各自所需的迭代次数及达到精度 $E_k < 10^{-5}$ 所需的运算时间. 进一

步, 为证明 GS-LSPIA 的收敛速度, 图 8 显示了例 4~例 6 的迭代次数与拟合误差的关系图. 从表 2 和图 8 可以看出, GS-LSPIA 算法的收敛速度要快于 LSPIA 算法.

表 2 图 5~图 7 中数据点集在拟合精度 E_k 下的迭代次数及时间

模型	算法	迭代次数					时间/s
		$E_k < 10^{-1}$	$E_k < 10^{-2}$	$E_k < 10^{-3}$	$E_k < 10^{-4}$	$E_k < 10^{-5}$	
多峰曲面	LSPIA ^[9]	1	4	21	59	146	0.523 03
	GS-LSPIA	1	3	9	17	34	0.369 31
壳体曲面	LSPIA ^[9]	0	1	5	22	64	1.629 31
	GS-LSPIA	0	1	2	6	19	1.422 16
Tranguloid 三叶曲面	LSPIA ^[9]	0	1	8	33	86	3.226 34
	GS-LSPIA	0	1	4	13	27	2.713 72

注. 时间为达到精度 $E_k < 10^{-5}$ 所需的运算时间.

图 8 不同算法的迭代次数与拟合误差关系图

3 结 语

本文提出一种基于 Gauss-Seidel 迭代方法的快速 PIA 算法, 称为 Gauss-Seidel 最小二乘几何迭代法(GS-LSPIA). GS-LSPIA 算法通过迭代调整控制点来生成一系列拟合曲线(曲面), 且极限曲线(曲面)是对给定数据点的最小二乘拟合结果. 大量实验结果证明了 GS-LSPIA 算法的有效性, GS-LSPIA 算法的收敛速度快于经典的 LSPIA 算法. 广义 B 样条基(generalized B-spline basis, GB)是对经典 B 样条基的扩展, 它用单调递增的连续可微非线性核函数代替经典的线性核函数. GB 在实际应用中更灵活. 因此, 后续工作会将 GS-LSPIA 与 GB 结合, 加快 GS-LSPIA 的收敛速度.

参考文献(References):

- [1] Qi Dongxu, Tian Zixian, Zhang Yuxin, *et al.* The method of numeric polish in curve fitting [J]. *Acta Mathematica Sinica*, 1975, 18(3): 173-184(in Chinese)
(齐东旭, 田自贤, 张玉心, 等. 曲线拟合的数值磨光方法 [J]. *数学学报*, 1975, 18(3): 173-184)
- [2] de Boor C. How does Agee's smoothing method work?[R]. Washington D C: Army Research Office, 1979
- [3] Lin H W, Wang G J, Dong C S. Constructing iterative non-uniform B-spline curve and surface to fit data points[J]. *Science in China: Series F*, 2004, 47(3): 315-331
- [4] Lin H W, Bao H J, Wang G J. Totally positive bases and progressive iteration approximation[J]. *Computers & Mathematics with Applications*, 2005, 50(3): 575-586
- [5] Delgado J, Peña J M. Progressive iterative approximation and bases with the fastest convergence rates[J]. *Computer Aided Geometric Design*, 2007, 24(1): 10-18
- [6] Shi Limin, Wang Renhong. An iterative algorithm of NURBS interpolation and approximation[J]. *Journal of Mathematical Research and Exposition*, 2006, 26(4): 735-743(in Chinese)
- (史利民, 王仁宏. NURBS 曲线曲面拟合数据点的迭代算法 [J]. *数学研究与评论*, 2006, 26(4): 735-743)
- [7] Lin H W. Local progressive-iterative approximation format for blending curves and patches[J]. *Computer Aided Geometric Design*, 2010, 27(4): 322-339
- [8] Lin H W, Zhang Z Y. An extended iterative format for the progressive-iteration approximation[J]. *Computers & Graphics*, 2011, 35(5): 967-975
- [9] Deng C Y, Lin H W. Progressive and iterative approximation for least squares B-spline curve and surface fitting[J]. *Computer-Aided Design*, 2014, 47(1): 32-44
- [10] Lin H W, Cao Q, Zhang X T. The convergence of least-squares progressive iterative approximation for singular least-squares fitting system[J]. *Journal of Systems Science and Complexity*, 2018, 31(6): 1618-1632
- [11] Lu L Z. Weighted progressive iteration approximation and convergence analysis[J]. *Computer Aided Geometric Design*, 2010, 27(2): 129-137
- [12] Zhang L, Ge X Y, Tan J Q. Least square geometric iterative fitting method for generalized B-spline curves with two different kinds of weights[J]. *The Visual Computer*, 2016, 32(9): 1109-1120
- [13] Zhang L, Tan J Q, Ge X Y, *et al.* Generalized B-splines' geometric iterative fitting method with mutually different weights[J]. *Journal of Computational and Applied Mathematics*, 2018, 329: 331-343
- [14] Liu M Z, Li B J, Guo Q J, *et al.* Progressive iterative approximation for regularized least square bivariate B-spline surface fitting[J]. *Journal of Computational and Applied Mathematics*, 2018, 327: 175-187
- [15] Ebrahimi A, Loughmani G B. A composite iterative procedure with fast convergence rate for the progressive-iteration approximation of curves[J]. *Journal of Computational and Applied Mathematics*, 2019, 359: 1-15
- [16] Chen Z X, Luo X N, Tan L, *et al.* Progressive interpolation based on catmull-Clark subdivision surfaces[J]. *Computer Graphics Forum*, 2008, 27(7): 1823-1827
- [17] Cheng F H, Fan F T, Lai S H, *et al.* Loop subdivision surface based progressive interpolation[J]. *Journal of Computer Science and Technology*, 2009, 24(1): 39-46
- [18] Deng C Y, Ma W Y. Weighted progressive interpolation of loop subdivision surfaces[J]. *Computer-Aided Design*, 2012, 44(5): 445-455

- 424-431
- [19] Wang Z H, Li Y J, Ma W Y, *et al.* Gauss-seidel progressive iterative approximation (GS-PIA) for loop surface interpolation[C] //Proceedings of Pacific Conference on Computer Graphics and Applications - Short Papers. Aire-la-Ville: Eurographics Association Press, 2018: 73-76
- [20] Liu Xiaoyan, Deng Chongyang. Non-uniform cubic B-spline curve interpolation algorithm of GS-PIA[J]. Journal of Hangzhou Dianzi University: Natural Sciences, 2015, 35(2): 79-82(in Chinese)
(刘晓艳, 邓重阳. 非均匀三次 B 样条曲线插值的 GS-PIA 算法[J]. 杭州电子科技大学学报: 自然科学版, 2015, 35(2): 79-82)
- [21] Zhang Y F, Wang P, Bao F X, *et al.* A single image super-resolution method based on progressive-iterative approximation[J]. IEEE Transactions on Multimedia, 2020, 22(6): 1407-1422
- [22] Hamza Y F, Lin H W, Li Z H. Implicit progressive-iterative approximation for curve and surface reconstruction[J]. Computer Aided Geometric Design, 2020, 77: 101817
- [23] Golub G H, van Loan C F. Matrix computations[M]. 3rd ed. Baltimore: Johns Hopkins University Press, 1996
- [24] Piegl L, Tiller W. The NURBS book[M]. 2nd ed. Heidelberg: Springer, 1997