

几何迭代法的加速

刘 超¹⁾, 辛士庆¹⁾, 舒振宇²⁾, 陈双敏^{1)*}, 张 荣¹⁾, 赵杰煜¹⁾

¹⁾ (宁波大学信息科学与工程学院 宁波 315211)

²⁾ (浙江大学宁波理工学院信息科学与工程学院 宁波 315211)
(chenshuangmin@nbu.edu.cn)

摘 要: 考虑到传统的几何迭代法仅有一阶的收敛性, 提出一个二阶可导的能量函数来刻画当前曲线与目标点集之间的差异. 首先根据初始的控制顶点和相应的基函数生成初始的样条曲线, 然后求差异函数关于各个控制顶点的梯度, 最后采用 L-BFGS 算法快速寻找最优的插值或者逼近曲线. 实验结果表明, 文中算法具有超线性的收敛速度, 在同样的精度要求下比原来的几何迭代法快出数十倍甚至上百倍; 既可用于插值问题, 也可用于逼近问题; 甚至也能适用于数据点参数可变的情形.

关键词: 几何迭代法; L-BFGS 算法; 插值; 逼近
中图法分类号: TP391.41

Accelerated Geometric Iteration Method

Liu Chao¹⁾, Xin Shiqing¹⁾, Shu Zhenyu²⁾, Chen Shuangmin^{1)*}, Zhang Rong¹⁾, and Zhao Jieyu¹⁾

¹⁾ (Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211)

²⁾ (School of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University, Ningbo 315211)

Abstract: Considering that the conventional geometric iteration method (GIM) has only 1-order convergence, this paper proposes an energy function with second-order smoothness that characterizes the difference between the up-to-date curve and the given data points. In numerical implementation, we initialize the spline curve according to the initial control points and the associated basis functions, then compute the difference function as well as the gradients with regard to the moveable control points, and finally use the L-BFGS technique to find the optimal interpolation/approximation curve. Experimental results show that our accelerated algorithm has a super-linear rate of convergence. With the same accuracy requirement, the improved GIM outperforms the original version by tens to hundreds of times in terms of efficiency. It can be used to both the interpolation problem and the approximation problem, and even to the case where the input data points have variable parameters.

Key words: geometric iteration method; L-BFGS algorithm; interpolation; approximation

几何迭代法(geometric iteration method, GIM)的思想最早可追溯到我国 CAGD 领域的前辈齐东

旭教授^[1]于 1975 年提出的针对均匀三次 B 样条曲线的盈亏修正算法. 给定一个数据点列, 其中每个

收稿日期: 2016-05-07; 修回日期: 2016-08-15. 基金项目: 国家自然科学基金(61300168, 11226328, 61571247); 浙江省自然科学基金(LY13F020018, LZ16F030001); 浙江省重中之重开放课题(XKXL1429, XKXL1406); 宁波大学研究生教育教学研究项目(JGZDI201501); 宁波大学教研项目(JYXMxsj201405). 刘 超(1989—), 男, 硕士研究生, 主要研究方向为计算机图形学; 辛士庆(1979—), 男, 博士, 副教授, 硕士生导师, CCF 会员, 主要研究方向为计算机图形学、计算几何; 舒振宇(1979—), 男, 博士, 副教授, 主要研究方向为计算机图形学、数字几何处理、机器学习; 陈双敏(1982—), 女, 博士研究生, 助教, 论文通讯作者, 主要研究方向为深度学习与特征提取; 张 荣(1970—), 女, 博士, 副教授, 硕士生导师, 主要研究方向为计算机视觉, 数字图像取证; 赵杰煜(1965—), 男, 博士, 教授, 博士生导师, CCF 会员, 主要研究方向为机器学习与人工智能.

点被赋予一个固定的参数, 盈亏修正算法从一条初始的混合曲线开始, 通过不断地调整控制顶点, 减小点列与曲线之间的差距, 最终插值或者逼近给定的数据点列.

de Boor 算法^[2]针对用均匀三次 B 样条曲线逼近点集的情形, 证明了盈亏修正算法的收敛性. 2004 年, Lin 等^[3]证明了非均匀三次 B 样条曲线曲面同样具有这样的盈亏修正性质; 2005 年, Lin 等^[4]又证明了所有的全正基混合曲线曲面都具有这种几何性质, 并用渐近迭代逼近(progressive-iterative approximation, PIA)来描述这一算法.

2007 年, 日本学者 Maekawa 等^[5]建议对数据点的参数取值放松要求, 在每次迭代中都将数据点的参数更新为曲线曲面上最近点的相应参数值, 并将它命名为几何插值法(geometric interpolation, GI). 2011 年, Lin 等^[6]提出了逼近型的 PIA 算法, 通过将每一步迭代中的数据点差向量分组对应到相应控制点上, 得到控制点差向量实现曲线逼近, 称为 EPIA 算法. 2014 年, Deng 等^[7]设计了迭代极限是最小二乘解的 PIA 算法, 将数据点与当前曲线的差向量反映到各个控制顶点上去, 迭代步骤与 EPIA 相同, 被称为 LSPIA 算法. 最近, 蔺宏伟^[8]综述了盈亏修正算法、渐近迭代逼近以及几何插值法的相关工作, 并用 GIM 来统称这类算法.

GIM 现已被广泛应用到很多领域. 例如, 在几何设计领域中, 被用于计算有理三角 B-B 曲面的多项式逼近曲面^[9], 插值给定位置, 切向量和曲率向量的 GIM^[10-12], 数据的自适应拟合^[13]. 在逆向工程领域中, 实现曲线^[14]和曲面^[15]重建. 在体网格中, 生成有质量保持的六面体网格^[16]. 在工程实践领域中, 被应用到诸如汽车车灯、涡轮发动机、机翼几何外形设计等; 还被用于卫星图像处理^[17]、模式识别^[18]、手绘曲线逼近^[19]、有理曲线逼近^[20]、URBS 水印^[21], 以及树干造型^[22]等其他应用.

GIM 有以下优点: 1) 有明显的几何意义, 通过缩小数据点集与当前曲线曲面的距离来调整控制顶点, 最终能够达到插值或者逼近的目的; 2) 根据给定问题的不同, 易于在迭代过程中加入适当的约束, 使得极限曲线曲面仍然能够满足这些约束条件; 3) 收敛性有保证.

本文通过大量的实验发现, GIM 仅有一阶的收敛性, 并且其实际收敛速度与基函数形成的配置矩阵有关. 因此, 本文遵循 GIM 的基本思想, 提出一个二阶可导的能量函数(假设基函数也有二阶光

滑性), 用于刻画当前曲线与目标点集之间的总体差异. 在实际求解时, 结合目标函数的梯度信息采用 L-BFGS 算法求其最优解. 实验结果表明, 该算法能够收敛到与原来算法相同的结果, 但极大地减少了迭代次数, 从而在效率上获得了数十倍甚至上百倍的提高.

1 GIM 的基本思想

给定一个数据点列 $\{Q_i, i = 0, 1, \dots, n\}$, 其中每个数据点 Q_i 被赋予一个参数值 $t_i, i = 0, 1, \dots, n$, 且满足 $t_i < t_1 < L < t_n$. 求解以 $\{B_0(t), B_1(t), \dots, B_n(t)\}$ 为全正基底的混合曲线, 插值这些数据点.

以 $\{Q_i, i = 0, 1, \dots, n\}$ 为初始的控制点列, 构造一条初始混合曲线

$$P^{(0)}(t) = \sum_{i=0}^n P_i^{(0)} B_i(t),$$

其中, $P_i^{(0)} = Q_i$ 为数据点列中的第 i 个点, $B_i(t)$ 为基函数, $i = 0, 1, \dots, n$. 假设第 k 次迭代后生成的混合曲线

$$P^{(k)}(t) = \sum_{i=0}^n P_i^{(k)} B_i(t)$$

为进行第 $k+1$ 次迭代, 首先计算差向量

$$\Delta_i^{(k)} = Q_i - P^{(k)}(t_i), i = 0, 1, \dots, n;$$

然后将其加到曲线 $P^{(k)}(t)$ 的相应控制顶点上, 即

$$P_i^{(k+1)} = P_i^{(k)} + \Delta_i^{(k)}, i = 0, 1, \dots, n;$$

从而生成第 $k+1$ 次曲线

$$P^{(k+1)}(t) = \sum_{i=0}^n P_i^{(k+1)} B_i(t).$$

按此规律迭代, 可产生一个曲线序列

$$\{P^{(k)}(t), k = 0, 1, 2, \dots\}.$$

记在第 k 次迭代中产生的差向量为

$$\Delta^{(k)} = [\Delta_1^{(k)}, \Delta_2^{(k)}, \dots, \Delta_n^{(k)}]^T,$$

则易知

$$\Delta^{(k+1)} = (I - C)\Delta^{(k)};$$

其中, I 为单位矩阵, C 为基函数的配置矩阵, 即

$$C \triangleq \begin{bmatrix} B_0(t_0) & B_1(t_0) & \cdots & B_n(t_0) \\ B_0(t_1) & B_1(t_1) & \cdots & B_n(t_1) \\ \vdots & \vdots & \vdots & \vdots \\ B_0(t_n) & B_1(t_n) & \cdots & B_n(t_n) \end{bmatrix}.$$

文献[4]指出, $\Delta^{(k)}$ 的递减与 $(I - C)$ 的谱半径有关; 更进一步, 上述迭代格式具有一阶的收敛性.

2 GIM 的加速技巧

尽管 GIM 有诸多优点,但考虑到传统的 GIM 只有一阶的收敛性,本文遵循 GIM 的基本思想,提出一个光滑的能量函数,用于刻画当前曲线与目标点集之间的差距;然后采用 L-BFGS 算法求其最优解。下面详细叙述能量函数及梯度函数的具体形式,以及数值求解算法。

2.1 刻画差距的能量函数及梯度函数

GIM 的核心思想是最小化点集 $\{Q_i\}$ 与当前曲线

$P^{(k)}(t) = \sum_{i=0}^n P_i^{(k)} B_i(t)$ 的最小差距。因此,本文考虑二者之间的整体差距,定义如下的能量函数。

$$f(P_1^{(k)}, P_2^{(k)}, \dots, P_n^{(k)}) = \sum_{i=0}^n \left\| Q_i - P^{(k)}(t_i) \right\|^2 = \sum_{i=0}^n \left\| Q_i - \sum_{j=0}^n P_j^{(k)} B_j(t_i) \right\|^2 \left(\triangleq \|A_i^{(k)}\|^2 \right) \quad (1)$$

事实上,该能量函数刚好是差距向量 $A_i^{(k)}$ 的二范数的平方。式(1)关于变量 $P_j^{(k)}$ 的偏导数为

$$\frac{\partial f}{\partial P_j^{(k)}} = -2 \sum_{i=0}^n B_j(t_i) \left(Q_i - \sum_{h=0}^n P_h^{(k)} B_h(t_i) \right).$$

更进一步,函数 f 的 Hessian 矩阵为

$$\frac{\partial^2 f}{\partial P_j^{(k)} \partial P_i^{(k)}} = 2 \sum_{i=0}^n B_j(t_i) B_i(t_i) = 2C \times C^T,$$

其中 C 为配置矩阵。

基于上面的推导,本文可以用已有的数值算法对 GIM 进行加速,如牛顿法或者 L-BFGS。考虑到当问题的规模非常大(数据点很多)时 Hessian 矩阵并不稀疏,采用牛顿法需要求 Hessian 矩阵的逆矩阵,将会占用较多的空间而且比较耗时。所以,本文建议采用 L-BFGS 加速 GIM,它仅需要提供目标函数及其梯度函数,同时又能保证超线性的收敛性。下面指出上述算法框架的三大特点。

1) 快速收敛性。根据上面的推导,当配置矩阵为可逆矩阵时, Hessian 矩阵正定;否则,为半正定矩阵(如当两数据点对应于同一个参数时)。无论哪种情况,能量函数都有基本唯一的最小值,采用经典的数值算法(牛顿法或者 L-BFGS)都能保证其超线性收敛性。就本质而言,传统的 GIM 相当于最小化残差的无穷大范数,而本文算法则是最小化残差的 2 范数的平方。显然,后者对应的代价函数具有更好的光滑性,这意味着可以在优化方面具有更大的施展空间和更快的收敛性。

2) 逼近与插值的可切换性。特别地,当数据点数目与控制顶点数目一致时,配置矩阵 C 一般情况下可逆,上述 Hessian 矩阵正定,因而最优解唯一, L-BFGS 算法能够找到该类问题的最优解(即插值曲线)。当数据点明显多于控制顶点数目时, Hessian 矩阵仍然正定或者半正定,但 L-BFGS 算法仅依赖于目标函数和梯度信息,因而也能够找到(可能不唯一的)最优解,即逼近曲线。甚至,当数据点严重不足时插值曲线并不唯一,本文算法也能给出其中一个解。

3) 参数可变性。正如前面提到的, GIM 放松了参数固定不变这一要求,在每次迭代中都将数据点的参数更新为曲线曲面上最近点的相应参数值。本文算法也同样支持这一点。所不同的是,在这种情况下, $\{t_i\}$ 也是变量,梯度及 Hessian 矩阵应相应地做出改变;在迭代过程中, $\{t_i\}$ 并不是更新为最近点的参数,而是应该遵循数值算法的具体更新策略。

2.2 数值求解

本文采用 L-BFGS 算法^[23-29]求解能量函数的最优解。L-BFGS 算法只依赖能量函数及其梯度函数,可看作是共轭梯度法的扩展;所不同的是,它可以在优化能量函数的同时自动估算 Hessian 矩阵。在目标函数二阶光滑时, L-BFGS 算法一般具有超线性收敛性,并且 L-BFGS 算法节省内存,可适用于大规模优化问题的求解。

下面简要介绍 L-BFGS 算法的思想^[29]:用单位矩阵初始化 Hessian 矩阵的逆矩阵 \tilde{H}_0 ,确定 \tilde{H}_0 的修正次数 M 。此处, M 的取值一般为 3~20。本文用 x_k 代表第 k 次迭代中的控制顶点,用 g_k 表示第 k 次迭代时的梯度向量。第 k 次迭代的伪代码如下:

// g_k 表示第 k 次的梯度

Step1. 初始化. $r = -g_k$; 如果梯度小于给定的阈值,则退出循环。

Step2. L-BFGS 正向求解。

//一般 M 取值为 3~20

for $i = \min(M-1, k-1), \dots, 0$

$\gamma_i = \rho_i s_i^T r$;

$r = r - \gamma_i y_i$;

end

// \tilde{H}_0 在第 1 次迭代时选用单位矩阵

Step3. $d_k = \tilde{H}_k^0 r$

Step4. L-BFGS 逆向求解。

for $i = 0, \dots, \min(M-1, k-1)$

$d_k = d_k + s_i(\gamma_i - \rho_i y_i^T d_k)$;

end

// x_k 表示第 k 次迭代的自变量

Step5. 寻找满足 Wolfe 条件的 α_k , 使得

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \beta \alpha_k g_k^T d_k,$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \beta g_k^T d_k.$$

Step6. 用迭代中出现的 $\{s_i\}$ 和 $\{\rho_i\}$ 更新矩阵 \tilde{H}_m , 其中 $m = k + 1$.

Step7. 令 $x_{k+1} = x_k + \alpha_k d_k$, 转下一次迭代.

3 实验及结果分析

本文采用三次均匀 B 样条曲线插值或逼近相应的数据点. 更多类型的插值/逼近问题见文献[4].

3.1 实验配置

本文在配置为 2.4 GHz 英特尔双核处理器、内存 6 GB 的电脑上采用 C++ 编程实现了本文算法. 本文根据用户输入数据点的数目把参数域均分, 在此基础上进行迭代. 因为本质原理的不同, GIM 和本文算法无法采用完全相同的终止条件(尤其对于拟合问题). 因此, 本文对 GIM 的程序终止条件设为“相邻 2 次迭代的控制点之差的无穷大范数小于 $1E-6$ ”, 而 L-BFGS 算法的程序终止条件设为“梯度的模长小于 $1E-6$ ”. 最后同时显示 2 种算法的梯度变化.

3.2 本文算法的收敛性

本文在图 1a 所示的形状上比较了传统的 GIM 与本文算法的收敛速度. 该图有 9 个数据点, 要求找出通过这 9 个点的闭合三次 B 样条曲线. 为了达到梯度的模长小于 $1E-6$ 这一条件, 传统的 GIM 需要 134 次迭代, 耗时 266 ms; 而本文算法仅需要 8 次迭代, 耗时 92 ms. 从图 1b 中可以看出, 本文算法的梯度下降曲线更加陡峭, 具有超线性的收敛速度. 为了更深入地研究算法的收敛性, 本文做了大量的逼近实验, 如图 2~5 所示. 每幅图中显示了原始的数据点(用蓝色显示)、其中的 3 次迭代(控制网格用黑色显示、样条曲线用红色显示)、能量下降曲线和梯度下降曲线.

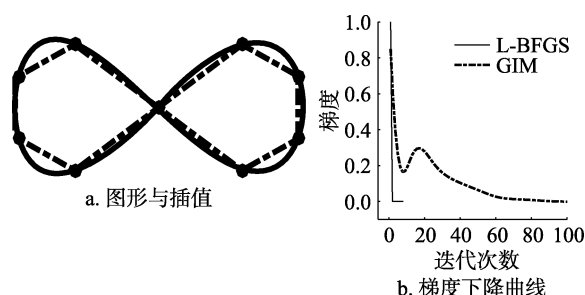


图 1 少量数据点的样条插值

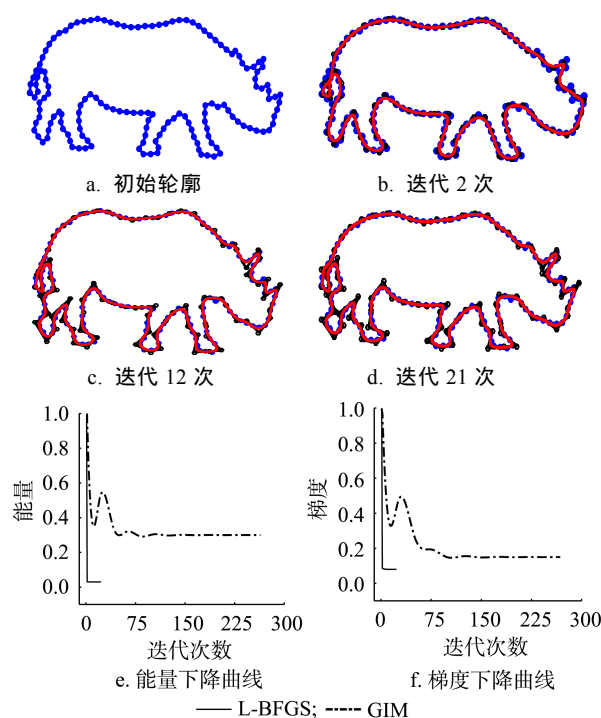


图 2 犀牛图形轮廓的逼近实验结果对比图

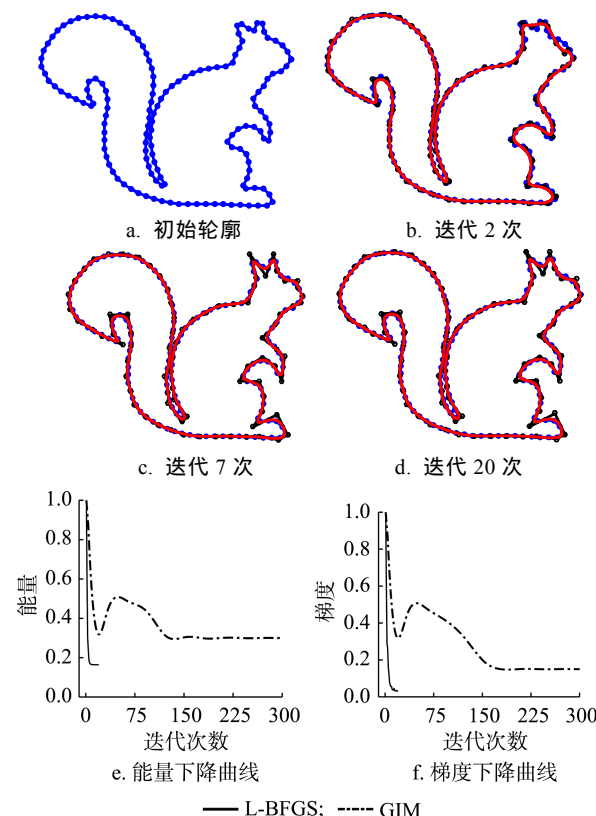


图 3 松鼠图形轮廓的逼近实验结果对比图

从图 2~5 中可以看出, 当数据点个数不超过 100 个时, 30 次迭代就足以达到指定的精度, 也就是 $1E-6$. 由于算法本身的超线性收敛性, 仅需额外

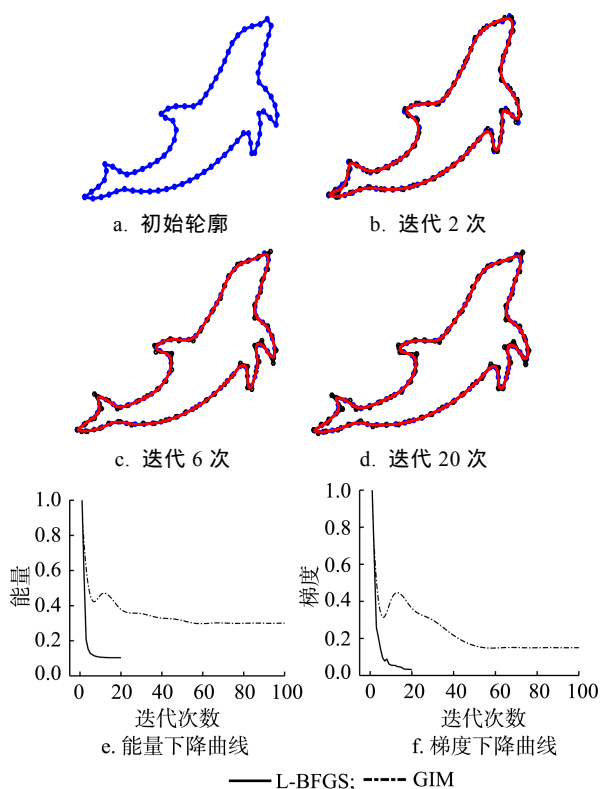


图4 海豚图形轮廓的逼近实验结果对比图

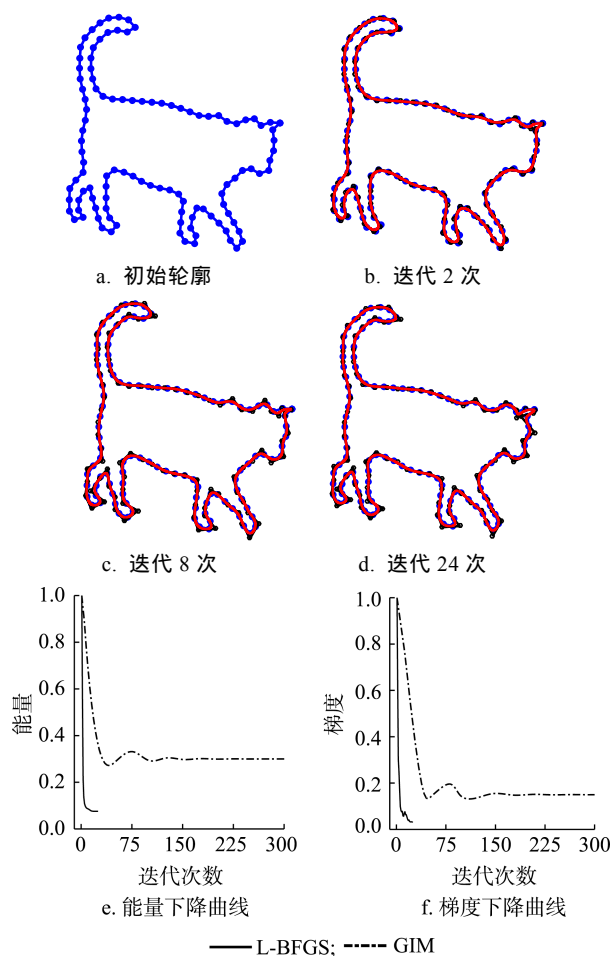


图5 小猫图形轮廓的逼近实验结果对比图

的少许迭代(10 次左右), 即可使梯度的模长小于 $1E-10$. 详细的时间代价如表 1 所示.

表 1 曲线逼近的迭代次数和时间

图号	数据点数/控制顶点数	算法名称	迭代次数	时间/ms
2	100/80	GIM	276	19834
		L-BFGS	23	2163
3	120/80	GIM	314	30312
		L-BFGS	20	2390
4	80/40	GIM	105	3281
		L-BFGS	20	906
5	110/83	GIM	297	22750
		L-BFGS	24	2250

事实上, 当问题的规模增大时, 传统 GIM 的迭代次数和运行时间显著增加, 而本文算法的优势则愈加明显. 之所以有更高的收敛速度, 主要是由“Hessian 矩阵的(半)正定性”决定的.

3.3 数据点参数可变的情形

本文算法具有更广的适用性和更大的灵活性, 允许数据点携带的参数值可变. 在具体实现时, 只需要将数据点的参数看作变量即可(两端的数据点例外). 如图 6 所示, 要求用 9 个控制顶点的三次 B 样条曲线逼近 15 个给定的数据点. 图 6a 所示为数据点参数值固定(均匀参数值)的情形, 而图 6b 所示为允许各数据点参数值可变的情况下得到的逼近曲线. 一般来说, 由于后者的解空间更大, 可以得到更好的逼近效果.

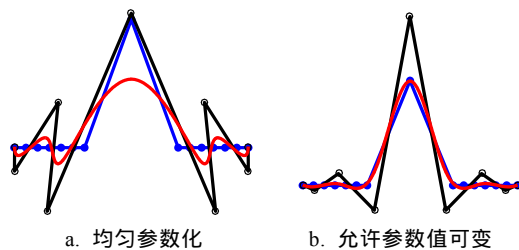


图6 数据点携带的参数值是否可变

4 结 语

考虑到传统的 GIM 具有明显的几何意义但收敛较慢的问题, 本文遵循 GIM 的思想, 提出了描述残差的能量函数. 由于它的 Hessian 矩阵正定, 本文建议用 L-BFGS 算法加速求解.

与传统的 GIM 相比, 本文算法具有三大优势: 快速收敛性, 逼近与插值的可切换性以及允许

数据点携带的参数值可变。

今后, 我们将把本文算法用于更多类型的曲线曲面插值/逼近问题。

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] Ding R. Drawing ruled surfaces using the dual de Boor algorithm[J]. *Electronic Notes in Theoretical Computer Science*, 2002, 61(4): 178-190
- [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: Information Sciences*, 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(3/4): 575-586
- [5] Maekawa T, Matsumoto Y, Namiki K. Interpolation by geometric algorithm[J]. *Computer-Aided Design*, 2007, 39(4): 313-323
- [6] Lin H W, Zhang Z Y. An extended iterative format for the progressive-iteration approximation[J]. *Computers & Graphics*, 2011, 35(5): 967-975
- [7] 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: 32-44
- [8] Lin Hongwei. Survey on geometric iterative methods with applications[J]. *Journal of Computer-Aided Design & Graphics*, 2015, 27(4): 582-589(in Chinese)
(蔺宏伟. 几何迭代法及其应用综述[J]. *计算机辅助设计与图形学学报*, 2015, 27(4): 582-589)
- [9] Hu Q Q. An iterative algorithm for polynomial approximation of rational triangular Bézier surfaces[J]. *Applied Mathematics and Computation*, 2013, 219(17): 9308-9316
- [10] Gofuku S I, Tamura S, Maekawa T. Point-tangent/point-normal B-spline curve interpolation by geometric algorithms[J]. *Computer-Aided Design*, 2009, 41(6): 412-422
- [11] Okaniwa S, Nasri A, Lin H W, *et al.* Uniform B-spline curve interpolation with prescribed tangent and curvature vectors[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2012, 18(9): 1474-1487
- [12] Xing Rongsheng, Pan Rijng. PIA for uniform cubic B-spline curve interpolation with prescribed and tangent vector[J]. *Journal of Fujian Normal University: Natural Science Edition*, 2014, 30(1): 25-32(in Chinese)
(星蓉生, 潘日晶. 三次均匀 B 样条曲线插值数据点及其切矢的 PIA 算法[J]. *福建师范大学学报: 自然科学版*, 2014, 30(1): 25-32)
- [13] Lin H W. Adaptive data fitting by the progressive-iterative approximation[J]. *Computer Aided Geometric Design*, 2012, 29(7): 463-473
- [14] Lin H W, Chen W, Wang G J. Curve reconstruction based on an interval B-spline curve[J]. *The Visual Computer*, 2005, 21(6): 418-427
- [15] Liu Y Z, Fu H S, Ju L Y. Application research of iterative minor adjustment in reverse engineering[C] // *Proceedings of the International Conference on Mechatronics and Automation*. Los Alamitos: IEEE Computer Society Press. 2009: 622-626
- [16] Lin H W, Liao H W, Deng C Y. Filling triangular mesh model with all-hex mesh by volume subdivision fitting[R]. Hangzhou: Zhejiang University. 2012
- [17] Khobkhun B, Prayote A, Rakwatin P, *et al.* Rice phenology monitoring using PIA time series MODIS imagery[C] // *Proceedings of the 10th International Conference Computer Graphics, Imaging and Visualization*. Los Alamitos: IEEE Computer Society Press. 2013: 84-87
- [18] Senawongsa S, Dejdumrong N. An approach to Thai decorative pattern recognition using Bézier curve representation with progressive iterative approximation[C] // *Proceedings of the 10th International Conference Computer Graphics, Imaging and Visualization*. Los Alamitos: IEEE Computer Society Press, 2013: 46-49
- [19] Nuntawisuttiwong T, Dejdumrong N. Approximating handwritten curve by using progressive-iterative approximation[C] // *Proceedings of the 10th International Conference Computer Graphics, Imaging and Visualization*. Los Alamitos: IEEE Computer Society Press, 2013: 33-37
- [20] Chantakamo A, Dejdumrong N. Conversion of rational Bézier curves into non-rational Bézier curves using progressive iterative approximation[C] // *Proceedings of the 10th International Conference on Computer Graphics, Imaging and Visualization*. Los Alamitos: IEEE Computer Society Press, 2013: 38-41
- [21] Pan Z G, Sun S S, Zhang M M, *et al.* Watermarking NURBS surfaces[M] // *Lecture Notes in Computer Science*. Heidelberg: Springer, 2005, 3768: 325-336
- [22] Kuzelka K, Marusak R. Comparison of selected splines for stem form modeling: a case study in Norway spruce[J]. *Annals of Forest Research*, 2014, 57(1): 137-148
- [23] Shanno D F. On the convergence of a new conjugate gradient algorithm[J]. *SIAM Journal on Numerical Analysis*, 1978, 15(6): 1247-1257
- [24] Buckley A G. A combined conjugate-gradient quasi-Newton minimization algorithm[J]. *Mathematical Programming*, 1978, 15(1): 200-210
- [25] Liu D C, Nocedal J. On the limited memory BFGS method for large scale optimization[J]. *Mathematical Programming*, 1989, 45(1): 503-528
- [26] Nocedal J. Theory of algorithms for unconstrained optimization[J]. *Acta Numerica*, 1992, 1: 199-242
- [27] Qin Z W. The relationships between CG, BFGS, and two limited-memory algorithms[J]. *Furman University Electronic Journal of Undergraduate Mathematics*, 2016, 12(1): 5-20
- [28] Nash S G, Nocedal J. A numerical study of the limited memory BFGS method and the truncated-newton method for large scale optimization[J]. *SIAM Journal on Optimization*, 1991, 1(3): 358-372
- [29] Liu Y, Wang W P, Lévy B, *et al.* On centroidal voronoi tessellation-energy smoothness and fast computation[J]. *ACM Transactions on Graphics*, 2009, 28(4): Article No.101