线要素光滑算法的时间复杂度分析

朱 伟[1,2,3],沈 婕[1,2,3],郭立帅[1,2,3]

(1. 南京师范大学地理科学学院,江苏 南京 210023)
(2. 虚拟地理环境教育部重点实验室,江苏 南京 210023)
(3. 地理信息科学江苏省重点实验室,江苏 南京 210023)

[摘要] 线要素光滑算法是地图综合及地理信息可视化中的一类重要的算法,它可以实现线状地物的连续化表达,增强地图的表达效果. 随着高性能 GIS 乃至云 GIS 的飞速发展,如何在并行计算环境和云计算环境中提高线要素光滑算法的效率,满足人们对地图综合的效率和地理信息表达的实时性需求成为该领域发展的关键问题. 本文调研了常见的线要素光滑算法,按照拟合方式将算法分类,每一类中选取了一种代表性的算法,对其时间复杂度进行分析,并对这些算法的并行化进行了初步探讨.

[关键词] 线要素,光滑算法,时间复杂度,并行算法
[中图分类号] P208 [文献标志码] A [文章编号] 1001-4616(2012)04-0112-06

Analysis of the Time Complexity of Line Smoothing Algorithms

Zhu Wei[1,2,3],Shen Jie[1,2,3],Guo Lishuai[1,2,3]

(1. School of Geography Science,Nanjing Normal University,Nanjing 210023,China)
(2. Key Laboratory of Virtual Geographic Environment of Ministry of Education,Nanjing 210023,China)
(3. Key Laboratory of Geographic Information Science of Jiangsu Province,Nanjing 210023,China)

Abstract: Line smoothing algorithm is an important algorithm in map generalization and geo-information visualization. It can achieve a continuous representation of linear features,thus enhance the effect of map representation. With the rapid development of high-performance GIS and even cloud GIS,how to improve the efficiency of line smoothing algorithm in parallel and cloud environments,and to meet the demand of map generalization efficiency and real-time geo-information representation have become the key issues in this field. In this paper,the previous algorithms of line smoothing are reviewed and classified into four categories according to their fitting methods. One representative algorithm of each category is selected to be particularly analyzed for their time complexities. The way of parallelize these algorithms is also discussed.

Key words: line,smoothing algorithm,time complexity,parallel algorithm

线要素光滑是地图综合中的常用算子. 地图中的矢量线要素是由坐标点串对应的折线模拟表示,但是线要素会随坐标点间距离的加大表现出锯齿状折线,与现实中地理实体的光滑连续特征不相符合,并且严重影响地图的艺术表达效果. 因此,需要运用绘图技术把抽象出来的计算机离散数据进行光滑处理,使线数据形成连续的形态以增加图像的真实感[1].

针对线要素光滑算法,学者们从不同的方面进行了研究[2-6]. 张凤蛟[7]对曲线光滑过程中的拟合方法进行了研究; 徐庆荣[8]对插值过程中的步长进行了研究; 韩光瞬[9]等对光滑过程中出现的不合理的尖角提出了钝化方法; 赵博[10]等对五点光滑法、三次样条 Hermite 插值法和 B - 样条曲线法的光滑效果进行了对比. 王延亮[11]等对线性迭代光滑法进行了改进,解决了曲线不通过离散点的问题. 蕃正风[12]等提出一种近似斜轴抛物线加权平均插值法曲线光滑. 李云锦,钟耳顺[13]等通过长度比值点计算切矢量方向,结合改进算法的切矢量模计算方法,构造了一种近似的插值曲线. 孟雅琴[14]等对张力样条函数进行了研究和探讨. 这些研究多数是针对光滑的质量或对单个算法的改进,没有对光滑算法的时间复杂度做系统的分析和总结.

收稿日期:2012 - 05 - 23.
基金项目:国家自然科学基金(41071288、41171350).
通讯联系人:沈婕,博士,副教授,硕士生导师,研究方向:地图自动综合并行计算、电子地图与网络地图设计. E-mail:shenjie@njnu.edu.cn

当前,在计算机科学领域,网络集群、网格计算、高性能计算以及分布式并行计算等技术的快速发展推动了高性能并行 GIS 的发展[15]. 如何将传统的 GIS 算法引入到高性能并行计算环境中,建立响应速度快、计算能力强的 GIS 系统是高性能并行 GIS 发展的关键问题. 本文调研了一些常用的线光滑算法在串行环境下的执行方式和效率,并探讨了这些算法的并行适宜性.

1　常见的光滑算法及其分类

线要素光滑算法有很多种,且按照不同的分类标准这些算法有不同的分类结果. 本文调研了目前 GIS 系统中常用的 7 种光滑算法,并按照光滑过程中的拟合方法将这些算法分成了 4 类.

1.1　线要素光滑算法概述

常用的曲线光滑算法有线性迭代光滑法、正轴抛物线加权平均法、斜轴抛物线加权平均法、五点求导分段三次多项式插值法、三点求导分段三次多项式插值法、张力样条函数法、Bezier 曲线法等 7 种算法. 这些算法大体可以分为以下 2 种类型: 一是要求光滑后的曲线严格通过各离散点; 另一种则根据点列的大体趋势来拟合出一条大致的曲线,拟合后的曲线只要求符合离散点的大体形状和趋势而不要求经过离散点. 拟合曲线严格通过各离散点的光滑算法有: 正轴抛物线加权平均法、斜轴抛物线加权平均法、五点求导分段三次多项式、三点求导分段三次多项式、张力样条函数插值法; 拟合曲线不通过各离散点的光滑算法有: 线性迭代光滑法、Bezier 函数光滑法.

这些算法各有优缺点,其对比分析如表 1 所示:

表 1　常用曲线光滑算法优缺点

Table 1　Merits and demerits of common line simplification

算法	优点	缺点
线性迭代光滑法	计算简单; 易实现	光滑后的曲线向内收敛,不通过原离散点
正轴抛物线加权平均法	数学上严密,计算较简单; 光滑曲线通过各离散点	离散点稀疏时,相邻离散点之间的加权平均曲线会出现多余摆动; 最大曲率点可能会偏离离散点
斜轴抛物线加权平均法	最大曲率点在离散点上; 曲线通过各离散点; 两点间曲线较短	计算过程复杂
五点求导分段三次多项式插值法	数学上严密,计算简单; 局部控制性好; 整条曲线具有连续的一阶导数	对“之”字形连续迂回的曲线进行光滑时会产生自相交
三点求导分段三次多项式插值法	曲线通过各离散点; 整条曲线具有连续的一阶导数; 随着转角的变化自动改变松紧度,避免自相交	光滑度低
张力样条函数法	可控制张力系数; 曲线光滑效果较好; 方便灵活	所有数据必须同时参与计算,计算速度慢
Bezier 曲线法	控制曲线方便; 设计灵活	曲线不通过原始离散点; 局部控制能力弱; 曲线拼接不灵活

随着 GIS 应用领域的扩展,其处理的数据量越来越多,对精度和实时性的要求越来越高,现有的光滑算法中精度能达到要求的算法往往效率较低,而效率高的算法精度上又达不到要求,因此对该类算法的研究和改进势在必行.

除此之外,还有一些其他算法,本文搜集和整理的算法如表 2 所示:

表 2　线要素光滑算法

Table 2　Line simplification

序号	算法名称	来源	参数
1	线性迭代光滑法		迭代次数
2	正轴抛物线加权平均法[16]	徐庆荣,1993	插值步长
3	斜轴抛物线加权平均法[17]	毋河海,2004	插值步长
4	五点求导分段三次多项式插值算法		插值步长
5	三点求导分段三次多项式插值算法		插值步长
6	张力样条函数算法[18]	Schweikert,1966	张力系数,插值步长
7	Bezier 曲线光滑法		插值步长
8	切线抹角法[11]	王延亮,2005	张力系数
9	改进型切线抹角法[19]	王晓理,2010	尺度圆半径
10	近似斜轴抛物线加权平均插值法[12]	潘正风,1991	转角 α,插值步长
11	改进型斜轴抛物线加权平均法[13]	李云锦,2009	切矢量方向,切矢量模

1.2 基于拟合方法的线要素光滑算法分类

线光滑算法的实现过程中,通常要按照一定的方法对线要素进行拟合,然后根据拟合得到的曲线方程进行插值处理,从而达到光滑的效果. 为了便于分析和程序实现,本文按拟合方法把线光滑算法分为: 基于直线拟合的线光滑算法,基于抛物线拟合的线光滑算法,基于多项式拟合的线光滑算法和基于整体曲线拟合的线光滑算法.

(1) 基于直线拟合的线光滑算法

直线拟合是将离散点直接用直线连接起来,该类算法主要有线性迭代光滑法.

(2) 基于抛物线拟合的线光滑算法

抛物线拟合的基本原理是通过 3 点可以确定 1 条抛物线. 设抛物线方程为 $y = ax^2 + bx + c$,假设离散点坐标 $p_1(x_1, y_1)$, $p_2(x_2, y_2)$, $p_3(x_3, y_3)$, $\cdots$, $p_n(x_n, y_n)$,欲求 p_i 和 p_{i+1} 2 点之间的曲线方程时,要先根据 p_{i-1}, p_i, p_{i+1} 3 点求出 1 个抛物线方程 y_i,然后根据 p_i, p_{i+1}, p_{i+2} 3 点求出 1 个抛物线方程 y_{i+1},这样在重叠范围 p_i 和 p_{i+1} 2 点之间有 2 条二次曲线,最后用加权的方法获得的平均曲线即可作为 p_i 和 p_{i+1} 2 点之间的曲线. 该类算法主要有正轴抛物线加权平均法和斜轴抛物线加权平均法.

(3) 基于三次多项式拟合的线光滑算法

三次多项式拟合的基本原理是在相邻数据点间建立 1 个三次多项式曲线方程,曲线的光滑性是通过要求整条曲线具有连续的一阶导数来保证. 该类算法主要有五点求导分段三次多项式插值法和三点求导分段三次多项式插值法.

(4) 基于整体曲线拟合的线光滑算法

整体曲线拟合的基本原理是以所有的离散数据点为整体,拟合出 1 条光滑的曲线. 该类算法主要有张力样条函数插值法和 Bezier 曲线光滑法.

2 常见的光滑算法时间复杂度求解

一般而言,对于一个算法的分析主要是对算法效率的分析,包括时间复杂度分析和空间复杂度分析. 硬件技术的发展使得计算机的存储容量大大提高,存储空间的局限性对算法的影响大大降低. 因此,算法的时间复杂度分析是算法分析中的关键部分. 本文在分析算法时间复杂度时,先写出算法的伪代码,然后找出其基本语句与问题规模的关系,进而得到算法的时间复杂度. 本文将从上述光滑算法的分类中每一类选出一种具有代表性的算法,给出算法时间复杂度分析过程.

2.1 基于直线拟合的线光滑算法

基于直线拟合的线光滑算法有线性迭代光滑法. 线性迭代光滑法又叫"抹角法",该算法是建立在线性插值的基础上,通过一次又一次的迭代,最终达到曲线光滑的目的. 其算法原理如下(设平面上有 3 个点 A、B、C):

第一次插值算出 AB 和 BC 间的 1/4 处的点位,其为 1′,2′,3′,4′,连接 2′3′,就抹去了 B 点.

第二次插值是在 1′2′、2′3′、3′4′区间进行的,同样算出各区间 1/4 处的点位,其序号为 1″、2″、3″、…、6″,连接 2″3″和 4″5″,就抹去了 2′和 3′两点.

第三次、第四次的插值也是用类似的方法.

线性迭代光滑法的算法伪代码为:

其中 k 为迭代次数,n 为点数

```
①For i = 1 to k
②   For j = 2 to n - 1
③       insertPoint(Point[j - 1],Point[j],Point[j + 1])
```

该算法的基本操作是抹角函数 insertPoint(),该函数的时间复杂度为 $O(n)$,函数的执行频度为:

$$T(n) = \sum_{i=1}^{k} \sum_{j=2}^{n-1} 1 = k(n-2).$$

由于迭代次数 k 通常都是很小的,一般情况下取 4 就足够光滑了,所以该算法的时间复杂度为 $T(n) = O(n) \times O(1) = O(n)$.

2.2　基于抛物线拟合的线光滑算法

基于抛物线拟合的线光滑算法主要有正轴抛物线加权平均法和斜轴抛物线加权平均法. 本类算法选取正轴抛物线加权平均法进行分析. 正轴抛物线加权平均法又称为二次多项式平均加权法. 正轴抛物线加权平均法的基本思想是按数据点的顺序，每相邻 3 点作 1 条正轴抛物线. 设平面上有 $i-1$，i，$i+1$，$i+2$ 4 个数据点，是 n 个数据点的一部分. 通过 $i-1$，i，$i+1$ 3 点可以拟合 1 条二次曲线；通过 i，$i+1$，$i+2$ 3 点又可以拟合 1 条二次曲线，在重叠范围 i，$i+1$ 点之内有 2 条二次曲线，用加权的办法获得平均曲线作为最终的插值曲线.

正轴抛物线加权平均法的算法伪代码为：

其中 n 为线数据中点的总数

①Add Point [0] and Point [$n+1$]

②For $i=1$ to $n-1$

③　Parabola(Point [$i-1$]，Point [i]，Point [$i+1$])

④　Parabola(Point [i]，Point [$i+1$]，Point [$i+2$])

⑤　Curve(i to $i+1$) $= W_i \times$ Parabola(Point [$i-1$]，Point [i]，Point [$i+1$]) $+ W_{i+1} \times$ Parabola(Point [i]，Point [$i+1$]，Point [$i+2$])

该算法只有一个循环，步骤③、④、⑤均包含在循环②里面，步骤③、④、⑤的执行频度为：

$$T(n) = \sum_{i=1}^{n-1} 1 = n-1 = O(n).$$

步骤③、④为根据 3 点建立抛物线方程，时间复杂度均为 $O(1)$，步骤⑤是根据步骤③、④计算出的 2 条抛物线的重叠部分加权拟合曲线，时间复杂度是常数级，为 $O(1)$. 因此该算法的时间复杂度为 $T(n) = 3 \times O(n) \times O(1) = O(n)$.

2.3　基于三次多项式拟合的线光滑算法

基于三次多项式拟合的线光滑算法主要有五点求导分段三次多项式插值和三点求导分段三次多项式插值. 本类算法选取五点求导分段三次多项式插值法进行分析. 其核心思想是依次利用 5 个相邻的数据点，使用数学方法确定曲线在 5 点中第 3 点处的切线斜率，然后利用所求联立方程组，解出待定系数，确定曲线方程[20].

五点求导分段三次多项式插值法的算法伪代码为：

其中 n 为线数据中点的总数

①Add Point [-1]，Point [0]，Point [$n+1$] and Point [$n+2$]

②For $i=1$ to n

③　FirstDerivative(Point [i])

④For $i=1$ to $n-1$

⑤　Curve(Point [i]，Point [$i+1$])

该算法包含 2 个循环，分别为步骤②和步骤④，基本操作语句是步骤③和步骤⑤，步骤③的执行频度为：

$$T(n) = \sum_{i=1}^{n} 1 = n = O(n),$$

步骤⑤的执行频度为：

$$T(n) = \sum_{i=1}^{n-1} 1 = n-1 = O(n).$$

步骤③为根据 5 点求得中间点的导数，其时间复杂度为 $O(1)$，步骤⑤是根据多项式曲线方程拟合 2 点之间的部分曲线，时间复杂度为 $O(1)$. 因此该算法的时间复杂度为：

$$T(n) = O(n) \times O(1) + O(n) \times O(1) = O(n).$$

2.4　基于整体曲线拟合的线光滑算法

本类算法选取张力样条函数法进行分析. 其基本构思是分段插值函数为线性插值和 2 个双曲函数

$\mathrm{sh}\sigma x$ 和 $\mathrm{ch}\sigma x$ 的线性组合:$f(x)=c_1+c_2x+c_3\mathrm{sh}\sigma x+c_4\mathrm{ch}\sigma x$. 张力样条函数的一个主要特点就是有一个可变的张力系数. 当张力参数趋于零或趋于无穷大时的极限曲线,分别是等距节点的有理三次 B 样条曲线和其控制多边形,故张力参数可用于调节曲线的光顺性[14].

张力样条函数法的算法伪代码为:

其中 n 为点数

①For $i=1$ to n

②　GetAccumulateLength(Point [i])

③GetCoefficientOfTension()

④Calculate Second Derivate Of All Points

⑤For $j=1$ to $n-1$

⑥　Curve(Point [j],Point [j+1])

从上述伪代码可以看出,该算法包括 2 个循环,步骤①和步骤⑤,步骤②的执行频度为:

$$T(n)=\sum_{i=1}^{n}1=n=O(n),$$

步骤③和步骤④各执行一次,步骤⑥的执行频度为:

$$T(n)=\sum_{i=1}^{n-1}1=n-1=O(n).$$

步骤②为计算累计长度,其时间复杂度为 $O(1)$;步骤③为计算张力系数,其时间复杂度为 $O(n)$;步骤④计算二阶导数较复杂,在解二阶导数的过程中需要计算几个参数 Ai、Bi、Ci、Di,其时间复杂度均为 $O(n)$,求解 n 阶矩阵解二阶导数的时间复杂度也为 $O(n)$,所以步骤④的时间复杂度为 $4\times O(n)=O(n)$. 步骤⑥为根据张力样条函数拟合两点之间的曲线,其时间复杂度为 $O(1)$. 所以该算法的时间复杂度为:$T(n)=O(n)\times O(1)+O(n)+O(n)+O(n)\times O(1)=O(n)$.

3　分析与讨论

上节中计算出了 4 种线光滑算法的时间复杂度,利用同样的方式可以计算出其他光滑算法的时间复杂度. 本文计算了 7 种常用的光滑算法的时间复杂度. 计算结果如表 3 所示.

表 3　常用曲线光滑算法时间复杂度

Table 3　Time complexity of line smoothing algorithms

序号	算法	时间复杂度
1	线性迭代光滑法	$O(n)$
2	正轴抛物线加权平均法	$O(n)$
3	斜轴抛物线加权平均法	$O(n)$
4	五点求导分段三次多项式插值法	$O(n)$
5	三点求导分段三次多项式插值法	$O(n)$
6	张力样条函数法	$O(n)$
7	Bezier 曲线法	$O(n^2)$

从表 3 的分析可以看出,时间复杂度为线性的算法有 6 个:线性迭代光滑法、正轴抛物线加权平均法、斜轴抛物线加权平均法、五点求导分段三次多项式插值法、三点求导分段三次多项式插值法和张力样条函数法. 这类算法时间复杂度较低,在移植到并行计算环境下时,可以使用共享存储的方式来提高并行执行效率. 时间复杂度为非线性的算法有:Bezier 曲线法. 这类算法时间复杂度较高,可以考虑使用消息传递和共享存储混合的方式来提高算法在并行环境下的效率.

当要处理的数据量大,或者要执行的数据处理任务繁重,并且这些任务本身就可以分解为互不相关的子任务时,使用并行计算是合适的[21]. 在进行线光滑时,由于各条线要素间相互独立,因此光滑算法具有很好的并行性. 对单条线而言,线性迭代光滑法、正轴抛物线加权平均法和斜轴抛物线加权平均法、五点求导分段三次多项式插值法和三点求导分段三次多项式插值法进行光滑时都是按顺序从线要素中每次取出部分点进行操作,所以进行并行计算数据划分时可以对线段做分段处理. 为了保证分段点处曲线的连续性,在分段的时候要交叉分段,在合并结果数据的时候对交叉的数据段进行加权平均即可. 张力样条函数法计算各离散点二阶导数时要求解 n 阶矩阵,该过程可以并行化求解[22],在插值的时候是以 2 个点为单位计算张力样条函数进行插值,该过程可以把线段分段进行并行化求解. Bezier 曲线法也可以把线段分段进行并行化求解.

4　总结与展望

本文调研并整理了常用的线要素光滑算法，并对这些算法进行分类，然后从各个类别中选取具有代表性的算法，对其时间复杂度进行了具体分析和求解．基于分析的结果，将算法时间复杂度分为两类：线性算法和非线性算法，并探讨了适合这两类算法的并行化方法．这些探讨为线要素光滑算法并行计算奠定了基础．

本文仅对线要素光滑算法的时间复杂度进行了分析，对其并行化方法提出了初步设想．在线要素光滑算法高性能计算方面尚需做大量的研究工作．在空间数据特征面向线光滑算法对并行计算体系结构、并行算法设计等方面，还需进行实验设计与验证．

[参考文献]

[1] 王鹏．等值线快速绘图方法研究及系统设计与实现[D]．成都：电子科技大学通信学院，2011．

[2] Yu K. Performance improvement of Bezier spline fitting for more accurate approximation of natural linear entities[J]. KSCE Journal of Civil Engineering，1999，3(2)：181-193．

[3] 刘放，罗磊，胡俊，等．复杂轮廓曲线的样条插补与速度规划方法[J]．上海交通大学学报，2009，43(5)：834-836．

[4] Zeng Y，Nguyen T A，Yan B，et al. A distance-based parameter free algorithm for curve reconstruction[J]. Computer-Aided Design，2008，40(2)：210-222．

[5] 翟永强，刘正林．计算机制图中一种新的曲线光滑方法[J]．计算机工程与应用，2003(14)：93-95．

[6] Yue Y，Speckman P，Sun D. Priors for Bayesian adaptive spline smoothing[J]. Annals of the Institute of Statistical Mathematics，2012，64(3)：577-613．

[7] 张凤蛟．快速曲线拟合的方法[J]．延边大学学报：自然科学版，2006，32(3)：208-211．

[8] 徐庆荣．曲线插值中步长的确定[J]．武汉测绘学院学报，1983(1)：77-86．

[9] 韩光瞬，郭金丽．等高线光滑中不合理尖角钝化方法研究[J]．北京测绘，2010(4)：72-73．

[10] 赵博，谈俊仲．对 MapInfo 系统中线状要素光滑的研究[J]．测绘通报，2005(6)：25-27．

[11] 王延亮，王明爽．新型曲线光滑法——切线抹角法[J]．测绘通报，2005(3)：52-54．

[12] 潘正风，罗年学，黄全义．近似斜轴抛物线加权平均插值法曲线光滑[J]．测绘学报，1991，20(1)：60-65．

[13] 李云锦，钟耳顺，黄跃峰．斜轴抛物线插值的改进算法与近似算法[J]．武汉大学学报：信息科学版，2009，34(12)：1 490-1 494．

[14] 孟雅琴，叶正麟，王小平，等．空间均匀有理张力样条参数曲线[J]．计算机学报，2003，26(12)：1 776-1 780．

[15] 赵春宇．高性能并行 GIS 中矢量空间数据存取与处理关键技术研究[D]．武汉：武汉大学遥感信息工程学院，2006．

[16] 徐庆荣．计算机地图制图原理[M]．武汉：武汉测绘科技大学出版社，1993：110-115．

[17] 毋河海．地图综合基础理论与技术方法研究[M]．北京：测绘出版社，2004：54－61．

[18] Schweikert D. An interpolation curve using a spline in tension[J]. Journal of Mathmatical Physics，1966，2(45)：312-317．

[19] 王晓理．在线光滑法优化线要素多尺度表示[C]//信息工程大学测绘学院第五届博士生学术论坛论文集．郑州，2010．

[20] 张成岗，徐勇勇，陈长生．曲线拟合中的几个问题[J]．中国卫生统计，1994，11(2)：58-60．

[21] 金旭亮．NET 4.0 面向对象编程漫谈(应用篇)[M]．北京：电子工业出版社，2010：161．

[22] 孙世新，卢光辉，张艳，等．并行算法及其应用[M]．北京：机械工业出版社，2005：54－70．

[责任编辑：丁　蓉]