

大规模 DEM 数据并行可视域分析算法研究

亢晓丽¹, 亢晓琛^{2,3}

(1. 中国航空工业计算技术研究所, 西安 710068; 2. 武汉大学, 武汉 430079;

3. 中国测绘科学研究院, 北京 100830)

摘要: 可视域分析是具有高时间复杂性的地理计算, 在进行大规模数字高程模型 (DEM) 数据的可视域分析时, 传统的串行算法往往难以满足实际需求; 提出一种基于环形扫描环的并行可视域分析算法, 可以快速计算视点在大范围内的可视域范围, 与传统的扫描线算法相比, 并行环形扫描线算法以阵面方式向外扩张, 从而避免了各坐标点角度的重复计算; 通过试验对比发现, 并行环形扫描线算法的计算效率约为 ArcGIS 10.0 中可视域分析的 20 倍, 完全可以满足大规模 DEM 数据的可视域分析需求。

关键词: DEM; 环形扫描线; 可视域分析; 计算效率

Parallel Viewshed Analysis on Large Scale DEM Data

Kang Xiaoli¹, Kang Xiaochen^{2,3}

(1. AVIC Computing Technique Research Institute, Xi'an 710068, China; 2. Wuhan University, Wuhan 430079,

China; 3. Chinese Academy of Surveying and Mapping, Beijing 100830, China)

Abstract: Viewshed analysis is extremely computing intensive in geocomputation. Moreover, conventional serial algorithm are often difficult to meet this demand. We propose a parallel viewshed algorithm based on scanning circle which can compute the viewshed area quickly on large scale Digital Elevation Model (DEM). Compared with conventional algorithms, our algorithm computes the viewshed area in an expanded way which can help to avoid the redundant computation. Experiments demonstrate that our algorithm is 20 times faster than the viewshed analysis in ArcGIS 10.0, and can meet the demand of viewshed analysis on large scale DEMs.

Key words: DEM; scanning circle; viewshed analysis; computing efficiency

0 引言

可视域是指某一特定位置的可视范围。可视域分析是通过分析数字高程模型 (Digital Elevation Model, DEM) 数据中每个单元的高程值来确定到某些特定位置 (视点) 的可见性^[1]。可视域分析是大多数地理信息系统 (Geographical Information System, GIS) 软件, 如 ArcGIS, GRASS GIS, SAGA GIS 等的基本功能之一, 已被广泛应用于选址、路经规划、导航、景观规划以及环境变化监测^[2]。随着对地观测技术的快速发展, 获取大尺度、高分辨率的 DEM 数据逐渐成为可能^[3]。典型的技术诸如机载激光雷达 (Airborne Light Detection And Ranging) 所获取的点云数据可用于快速生成高分辨 DEM 数据^[4]。基于大规模 DEM 数据的可视域分析已经成为传统串行处理算法所面临的一个挑战。

当前, 计算机硬件技术得到快速发展。多核计算、集群计算及 GPU 计算成为提高传统算法效率的有效技术手段, 并被广泛采用。在单机环境下, OpenMP^[5] 和 Intel TBB^[6] 等线程级并行库的出现为开发高效算法提供了方便有效的技术支持。

Wu 等人^[3] 提出一种基于分块思想的串行可视域算法 (以下简称分块可视域算法), 该算法支持在大规模 DEM 数据基础上生成可视域。在分块可视域算法基础上展开研究, 本文提

出一种并行环形扫描算法。通过并行化其中最耗时的计算来提高算法效率。

1 并行环形扫描算法

并行环形扫描算法基于串行分块可视域算法 (1.1) 进行设计。通过分析算法中潜在的并行性, 实现算法的并行化 (1.2)。1.3 节从技术角度介绍了算法的实现。

1.1 串行分块可视域算法

串行分块可视域算法以一个参考平面算法作为基础。参考平面经过视点, 以及辅助网格上的同一行或者列上的一对参考点。参考平面算法的基本原则是: 可视的目标点必须在参考平面上或者高于该平面。这一对参考点位于目标点与视点之间, 且紧邻目标点。参考点的确定如图 1 所示。连接视点与目标点形成视线, 该视线与辅助网格最临近目标点的交点的两个端点即为参考点的位置。

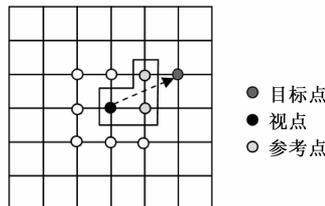


图 1 参考平面图

收稿日期: 2014-02-18; 修回日期: 2014-04-17。

作者简介: 亢晓丽 (1981-), 女, 山西人, 工学硕士, 主要从事计算机应用、无线通信网络和地理信息系统等方向的研究。

在得到参考点位置后, 依据视点与参考点来解算空间平面方程系数。代入三点坐标至公式 (1):

$$A(x) + B(y) + C(z) + D = 0 \quad (1)$$

计算得到平面方程系数后, 代入目标点的坐标位置, 即可得到目标点的可视高程。如果目标点的实际高程等于或者高于可视高程, 则视为可视点, 否则不可视。如果两个参考点重合(在视线方向为上、下、左、右、左上、左下、右上与右下方), 则令系数 C 为 0, 以直线方程来求解。目标点的选取围绕视点进行, 由内及外形成多个外环。在判别中, 需记录当各目标点对应的实际高程与可视高程二者中的最大值。这样, 在各环的判别中, 只需要与上一内环进行比较即可。

在参考平面算法基础上, 将数据进行分块。然后从视点所在块开始, 由内及外逐层处理。在分块模式下, 单次加载至内存的数据量可以进行动态控制, 从而有效降低内存使用。但是, 频繁的硬盘读写将会对算法效率造成严重的负面影响。此外, 内环单元格与外环单元格的计算存在依赖性, 在分块模式下难以并行进行改造。在具备大容量内存优势前提下, 完全可以将数据一次读入内存进行处理。基于该思路, 作者设计了并行环形扫描环算法, 一次读取所有数据用于并行处理。

1.2 并行环形扫描环算法

环形扫描算法从视点单元开始, 逐层扩展判别外层高程单元是否可视。第一层外环包括了视点单元外侧的八个方向(上, 下, 左, 右, 左上, 左下, 右上与右下方)的高程单元。由于邻接视点单元, 故各单元均为可视单元。第二层外环对应的高程单元的可视性由第一层外环单元与视点单元的可视角度决定。依次类推, 第 n 层外环由第 $n-1$ 层外环及之前的高程单元的可视角度决定。环形扫描环算法的计算依赖关系网如图 2 所示。

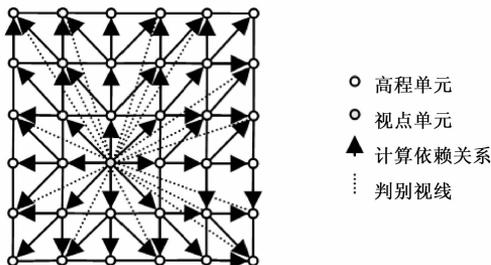


图 2 环形扫描线的依赖关系

在图 2 中, 外环高程单元的可视判别依赖于内环单元。计算视点的可视域范围时, 应该由内及外逐层判别。由视点向待判别单元发射视线, 取最后相交的位置所对应的高程单元位判别依据。在视点外侧 8 个方向上, 相交位置只有一个单元高程, 其它情况均会对应两个单元高程。由视点, 相交单元高程可构成一个平面。根据该平面可以计算目标单元所对应的平面高程, 即可视高程。高于可视高程的目标点将被标记为可视点, 低于可视高程的点则被标记为不可视点。从视点出发, 不同环上的单元高程的计算情况如图 3 所示。

各环的单元高程个数逐渐增加, 呈现 8 倍关系, 即 $M=8 * n$ 。同一环上的各单元高程的判别相对独立, 不存在依赖关系。因此, 可以将第 n 环的 M 个单元同时处理。

并行化可视域分析伪代码

初始化并行线程数 P , 读取实际高程列表 FH , 可视高程列表 VH ,

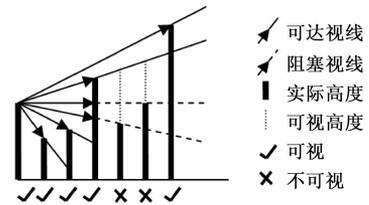


图 3 可视分析原理

可视性列表 V

根据视点计算单元高程环数 N ,

For $n = 1$ to $n = N$

计算各环单元高程个数: $M(n) = 8 * n$

For $m = 1$ to $m = M(n)$

(1) 分配当前单元对应计算的归属线程: $p = m \% P$

(2) 在 p 中计算上一环对应的依赖单元高程

(3) 根据视点单元与依赖单元计算平面方程 $Ax + By + Cz + D = 0$ 的系数 (A, B, C, D)

(4) 根据 (A, B, C, D) 计算 m 点位置可视高度 h

IF $FH[m] > h$

$V[m] = 1$

$VH[m] = FH[m]$

ELSE

$V[m] = 0$

$VH[m] = h$

End IF

输出可视性列表 V

1.3 算法实现

从视点栅格开始, 逐层计算外层栅格的可视性。由于各外层栅格的可见性由内层栅格的可达视线与外层栅格的高差所决定, 因此需要一个辅助可视外环来记录当前计算的外环栅格所对应的可视高程。在计算各层外环的可视性之前, 应首先计算可视外环。可视外环提供了对内部单元格的高程的规约, 以作为下一层外环的计算依据。同一高程环上的各个单元的计算完全独立, 可以在循环中并行进行。在 OpenMP 支持下, 通过指定多个并行线程来动态选取统一外环上的单元格进行计算。

在进行并行循环计算之前, 需要预先分配两个与数据规模(长 * 宽)相等的内存空间, 以加快数据读写。两块内存空间分别用于存储原始数据与计算结果数据。在将数据全部加载入内存后, 首先计算视点所在的栅格, 然后开始执行最为耗时可视域计算。各个计算线程的计算结果相互独立, 可直接写入到结果内存空间中。在计算结束后再将整个结果数据一次性写入到磁盘文件。

2 试验

2.1 数据及测试环境

本程序基于 C++ 语言开发, 并在 Visual Studio 2010 环境进行编译。CPU 为 Intel (R) Xeon (R) E5645@2.40 GHz * 24 (超线程支持); 内存为 DDR3, 32 GB; 硬盘为 2 GB, 7 500 mps。

试验数据选取了部分全国的 90 m 分辨率的 DEM 数据。试验内容包括结果精度测试(试验 1)与算法效率测试(试验

2) 两部分内容。试验 1 选取了规模为 1433×1377 的数据集, 并在该范围内随机选取 5 个视点来进行测试。试验 2 选取了 5 组不同规模的数据进行可视域分析的时间测试, 并在相同测试条件下 (相同的可视点位置) 与 ArcGIS 计算时间进行对比。

2.2 试验结果

试验 1 为计算结果的精度测试。DEM 数据及 5 个视点位置如图 4 所示。对比测试结果如表 1 所示。

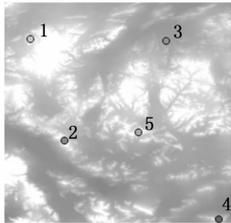


图 4 DEM 数据及视点位置

表 2 对比测试结果

组别	视点位置 (经度, 纬度)	可视点个数		误差(%)	计算时间(s)	
		ParaView	ArcGIS 10.0		ParaView	ArcGIS
1	82.238 9, 34.803 3	7 995	7 890	98.7	0.05	3.00
2	82.417 4, 34.273 2	94 891	94 169	99.2	0.03	3.00
3	82.942 2, 34.794 7	28 295	26 997	95.4	0.03	3.00
4	83.215 5, 33.864 4	210 929	203 557	96.5	0.03	3.00
5	82.798 8, 34.319 1	41 152	41 390	99.4	0.03	3.00

试验 2 为算法效率测试 (如图 5 所示), 分别针对 5 组不同规模的数据集。在试验中, 计算时间包括了数据读取时间, 数据处理时间以及结果输出时间。其中, ArcGIS 的计算时间为并行环形扫描算法的 20 倍左右。在处理第五组数据时, ArcGIS 的处理时间增加显著, 而并行扫描算法则仍旧保持了较好时间线性增长趋势。

2.3 讨论

从试验 1 的测试结果可知, 扫描环算法的计算结果精度与 ArcGIS 接近, 从而证明扫描环算法的可行性。在试验 2 中, 随着数据规模的增加, 扫描环算法的处理时间维持了线性增长趋势。计算效率明显优于 ArcGIS 中可视域分析算法。在处理第 5 组数据时, ArcGIS 的处理时间明显增加, 其根本原因是数据在内存与磁盘空间的频繁交换所致。

扫描环算法在处理大规模 DEM 数据时, 需要较大的物理内存来支持。从算法本身来看, 实时参与计算的数据只是整个数据范围内的一个小的局部。因此, 完全可以将数据计算与数据读写统一调度。从算法实现角度来看, 可以开辟两个独立的线程分别用于读取和写入数据。读取线程不断的从原始数据中读取邻接的外环, 并将其纳入待处理数据队列。多个并行处理线程则从数据列表中逐个取出数据环进行处理 (先进先出, 以保证计算依赖性), 并将计算得到的外环结果交给写入线程。

这样, 只要使得数据读取与数据计算的速度相匹配, 就可以在保证数据处理效率的情况下, 大大降低所需要的内存。

3 结论

计算机硬件技术已经得到极大的发展, 强大的多核计算能力以及廉价的内存空间为处理大规模 DEM 数据提供了可能。经典的分块可视域算法已经得到广泛应用, 本文在此基础上提出的并行环形扫描环算法。该算法在确保计算结果精确性的前提下, 可以充分发挥并行计算优势, 极大地改善大规模 DEM 数据的可视域分析的实际性能。与此同时, 作者提出了进一步优化该算法的技术思路。

参考文献:

[1] Jones E E. Using viewshed analysis to explore settlement choice: A case study of the Onondaga Iroquois. *American Antiquity*, 2006; 523-538.

[2] Kim Y H, Rana S, Wise S. Exploring multiple viewshed analysis using terrain features and optimisation techniques [J]. *Computers & Geosciences*, 2004, 30 (9): 1019-1032.

[3] Wu H, et al. A partition - based serial algorithm for generating viewshed on massive DEMs [J]. *International Journal of Geographical Information Science*, 2007, 21 (9): 955-964.

[4] Liu X, Peterson J, Zhang Z. High-resolution DEM generated from LiDAR data for water resource management [A]. *Proceedings of the International Congress on Modelling and Simulation (MODSIM05)* [C]. *Modelling and Simulation Society of Australia and New Zealand Inc.* 2005.

[5] Dagum L, Menon R. OpenMP: an industry standard API for shared-memory programming [J]. *Computational Science & Engineering*, IEEE, 1998, 5 (1): 46-55.

[6] Reinders J. Intel threading building blocks: outfitting C++ for multi-core processor parallelism [Z]. 2010; O'Reilly Media, Inc.

[7] Ahmed T. 7 Things You Must Know About Windows 7 32-Bit And 64-bit Version [Z]. 2010.

海泰电子推出交互式电子技术手册开发环境软件

陕西海泰电子有限责任公司近日发布了国内首款符合 GJB6600 标准的交互式电子技术手册 (Interactive Electronic Technical Manual, IETM) 开发环境软件, 简称 HTIETM DE。该软件对武器装备全寿命周期内产生、传递、使用的大量技术信息和数据提供系统的管理手段, 保证了工程资料、技术图纸、操作说明和维修数据的准确性、一致性、交互性和可操作性, 为用户提供了产品资料管理的完整解决方案。

HTIETM DE 以 GJB6600 标准为基准, 同时参考美军标和欧标的相关内容, 精心设计了创建、编辑、组织、发布、查看、更新升级等各种标准流程, 满足用户开发手册的工作习惯, 同时采用基于网络的 B/S 结构设计, 支持多用户团队跨部门/跨公司进行技术手册的协同开发、访问及部署, 突破了地域性的限制。HTIETM DE 采用模块化的开发方式, 将数据 (文字信息、表格、图片、音视频等) 封装成不同的模块, 不同的手册能够引用相同的模块, 提高了开发人员的工作效率, 降低了信息维护的成本与周期。