

一种面向并行空间数据库的数据划分算法研究

赵春宇¹ 孟令奎¹ 林志勇¹

(1 武汉大学遥感信息工程学院, 武汉市珞喻路 129 号, 430079)

摘要:面向基于对象关系型数据库而构建的并行空间数据库系统,提出了一种基于 Hilbert 空间填充曲线的适合于矢量空间数据的数据划分算法。在充分考虑空间信息的海量特征以及矢量数据存储记录的不定长等特点的前提下,该算法可实现并行空间数据库中海量空间数据记录在多个存储设备上的均衡划分,以避免出现数据倾斜现象,从而提高了空间数据的检索与查询效率。

关键词:并行空间数据库;数据划分;数据倾斜;Hilbert 空间填充曲线
中图分类号:P208

随着 GIS 应用的不断深入和拓展,尤其是多维动态 GIS 等多种应用类型的开发利用,用户对 GIS 的海量数据处理能力和效率提出了更高的要求,并行 GIS 及并行空间数据库应运而生。

并行空间数据库是并行 GIS 理论和技术研究的基础^[1]。并行数据库已有近 20 a 的发展历史,具有高性能、高可用性和高可扩展性等特点和优势,可满足于各种数据密集型业务的应用需求。然而,基于并行数据库系统构建空间数据库时将面临许多新的问题^[2-4]。本文针对并行空间数据库中数据划分技术进行研究,提出一种新的基于 Hilbert 空间填充曲线的空间数据划分方法,以实现海量空间数据在多磁盘系统上的较均匀分布,从而提高数据查询与检索的并行化程度。

1 空间数据划分算法设计

1.1 数据划分问题的提出

空间数据一般具有数据类型和存储结构多变、数据量巨大、空间关系复杂等特征,因此,采用对象关系型数据库实施对空间数据的组织与管理,其数据划分算法也有别于一般商业数据库中的数据划分策略。从数据特征来看,一般的商业关系数据和空间数据存在着较大的差别:由于使用的是常用数据类型,如整型、字符型等,一般商业关系数据库的每个元组大小是可以估算的,其上限是固定不

变的;而空间对象实体的存储是变长的,除点对象外,其他类型的空间对象对应的元组的大小均不相同。一般商业关系数据库的元组之间彼此孤立,而空间数据之间存在诸如拓扑关系、方位关系、度量关系等多种关联方式,在空间数据的划分过程中需要充分考虑空间关系因素的存在^[5]。

已有的数据划分算法均是针对一般关系型数据库而设计的,并没有考虑每个元组自身的数据量。这些算法强调的是关系中的元组在多个磁盘系统上的较均匀分布,即每个磁盘上所存储的元组的个数保持一个相对平衡的状态;而对于空间数据划分来说,元组的均匀分布很可能导致磁盘之间数据存储容量上的不均匀,从而产生严重的数据倾斜现象。

Oracle Spatial 模块是典型的使用对象关系型数据库进行空间数据管理的例子,提供了基于空间位置的范围划分策略,其中主要包括两种基本方法:基于 X 或 Y 坐标值的范围划分;基于 X 和 Y 坐标值的范围划分。这两种划分策略仅考虑到空间对象的位置属性,而没有考虑对象本身大小对数据划分产生的影响。假设某一个坐标范围内以点对象为主,而另一个坐标范围内分布较多的多边形对象。很明显,在这种情况下,这两个分区上的元组数目可能接近,但所存储的元组的总容量将相差非常大,产生了数据倾斜,从而导致多个磁盘间负载的不平衡。

并行 GIS 研究中更多的工作集中于并行索引机制的研究上^[6],而很少涉及空间数据的并行划分。例如, Kamel 和 Faloutsos 提出的采用空间填充曲线作为一维顺序码构造紧缩 R 树的方法,并给出了在多机系统之上的空间数据划分方法。当并发访问空间数据的用户增多时,这种方法将使主计算机成为系统访问瓶颈。同时,由于多数据集将不能按照相同的边界进行分区处理,所以该方法降低了针对多数据集的并行查询效率。本文针对以上问题,提出了并行空间数据库下的基于 Hilbert 空间填充曲线的空间数据划分方法 (spatial data partitioning based on Hilbert curve, HCSDP)。

1.2 HCSDP 算法设计

空间填充曲线是利用一个线性顺序来填充空间,可将多维空间中的对象映射到一维空间的一条曲线上,如 Z 曲线和 Hilbert 曲线等^[7,8]。由于 Hilbert 曲线具有优良的空间聚类特征^[9],因此,本文选择该曲线作为 HCSDP 算法的基础。

HCSDP 数据划分算法涉及的变量如下。

N : 系统中磁盘个数;

n : 待划分的空间实体的总数;

V_i : 第 i 个空间对象实体的大小,以 byte 为单位, i 的取值范围是 $[0, \dots, n-1]$;

V_{other} : 每个元组中所有非空间类型字段值的大小,因为除空间对象实体字段外,其他字段的大小是可以估计的,因而可以近似地用该变量统一表示,其单位为 byte;

V_{avg} : 每个磁盘上存储的所有空间实体的平均容量,以 byte 为单位;

B_j : 第 j 个桶中保存的空间对象实体总的大小,有 $B_j = B_j + V_k$ 成立,单位为 byte, k 的理论范围值是 $[0, \dots, n-1]$ 。每个桶对应一个磁盘, j 的取值范围是 $[0, \dots, N-1]$ 。

变量间有如下关系:

$$V_{\text{avg}} = \frac{nV_{\text{other}} + \sum_{i=0}^{n-1} V_i}{N} \quad (1)$$

算法步骤如下。

1) 扫描整个空间数据集,构造 Hilbert 曲线,为每个空间对象实体赋予 Hilbert 值,同时统计 V_i ,并计算 V_{avg} ,初始化 B_j 的值为零;

2) 根据 Hilbert 值,对空间实体进行排序;

3) 从 $i=0, k=0$ 开始,将第 k 个空间对象放入 $j=0$ 的桶中,并按 $B_j = B_j + V_k$ 计算 B_j 的值;

4) 比较 B_j 与 V_{avg} ,若 B_j 小于 V_{avg} ,则 $i=i+$

1, $k=k+1, j$ 不变;否则, $j=j+1, k=i$,重复此项操作,直至 $i=n-1, k=n-1, j=N-1$ 为止;

5) 如果执行 $B_{N-1} = B_{N-1} + V_k$, B_{N-1} 大于 V_{avg} ,而 $k < n-1$ 时,剩余的空间实体可采用轮转法,从标号为 N 的桶开始,以递减的顺序分配给各个桶;

6) 计算获得的所有 B_j 值,将这 N 个桶映射到 N 个处理节点上。

2 实验及分析

2.1 算法中 Hilbert 值生成方法描述

不失一般性,本文所选择的测试数据集的类型为多边形。由于 Hilbert 填充曲线是针对点对象而设计的,而要为多边形对象赋予 Hilbert 值,首先需要对数据集进行预处理。其方法是:计算获得每个多边形最小外接矩形的中心点 O_i ,以 O_i 表示第 i 个多边形;根据 O_i 的纵横坐标值,计算每个多边形对象的 Hilbert 值。通过该方法,可以将多边形数据集转换为点数据集,进而在新的点集之上构造 Hilbert 曲线。Hilbert 曲线构造函数的计算复杂度为 $O(m^2)$, m 为 Hilbert 曲线生成所需要进行的迭代运算的次数,或称为 Hilbert 曲线的次序。为尽可能地逼近点对象的真实位置,从而减少点对象集中的 Hilbert 值的重复次数。一般 m 值的选择需要根据待处理的空间对象的个数 n 来确定,即 $n < 2^m$,进而求解 m 值。考虑本文所选择的测试空间数据集及计算复杂度,令 $m=8$,即空间对象的 Hilbert 值经过 8 次迭代后获得,最多为 2^{16} 个空间实体。

2.2 实验结果对比分析

本文采用全国 1:100 万县市行政区域边界图作为测试数据集,共 2 330 个多边形地物,空间数据以 Shape 文件格式存储,大小分别为 17.6 Mb 和 8.2 Mb。为验证 HCSDP 算法的有效性,只使用 ID 字段域和几何对象字段域。ID 的类型为整型,即 V_{other} 为 4 byte;假设第 i 个空间对象由 m 个点组成,那么,其大小可近似表达为 $V_i=16m$,单位为 byte。其计算结果如表 1、表 2 所示。

表 1 3 个磁盘下数据划分情况

Tab. 1 Spatial Data Partitioning Based on 3 Disks

$V_{\text{avg}}=5\ 894\ 936$	B_0	B_1	B_2	总计
空间实体大小	5 894 841	5 894 878	5 895 089	17 684 808
空间实体个数	563	962	805	2 330

表2 5个磁盘下数据划分情况

Tab. 2 Spatial Data Partitioning Based on 5 Disks

$V_{avg} = 3\ 536\ 961.6$	空间实体大小	空间实体个数
B_0	3 536 911	377
B_1	3 536 945	409
B_2	3 536 988	558
B_3	3 536 979	403
B_4	3 536 985	583
总计	17 684 808	2 330

从表1、表2可以看出, HCSDP 算法可以实现空间数据集在多个磁盘上的均匀分布, 这里的均匀分布是指每个磁盘存储的数据大小相对均衡, 而不是指空间实体个数的相对平衡。从表中结果还可以看出, 每个磁盘所存储的空间实体大小相差不大, 但实体个数存在较明显的差别, 这与空间对象的复杂程度相关。

图1是本文所采用数据集在使用 Oracle Spatial 提供的空间数据划分方法进行划分处理后的结果示意图。

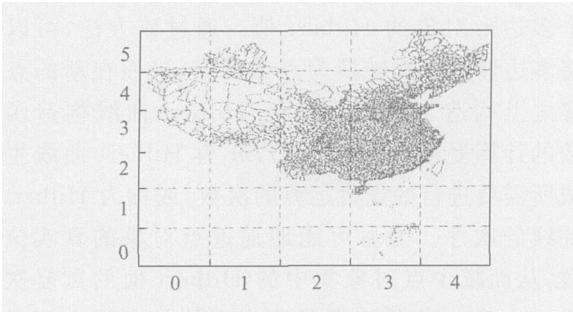


图1 Oracle Spatial 的数据划分策略

Fig. 1 Data Partitioning Strategy of Oracle Spatial

图1所示的数据划分策略类似于在数据集上构建格网索引的过程, 并采用轮转法将每个格网单元分别映射到多个存储节点上。其映射方法为: 首先为格网单元编号, 从0开始, 最大编号为 $MN - 1$, M 和 N 分别是横向与纵向的单元格数目; 然后, 利用传统的轮转法, 确定每个单元格所对应的磁盘的序号。从图1不难看出, 每个单元格所覆盖的空间对象的个数差别很大, 进而导致多磁盘间出现数据倾斜现象。实践证明, 即使采用不同的划分边界, 也很难避免数据倾斜现象的产生, 同时, 确定不规则格网边界所需要的计算也将增加数据划分算法的设计与实现难度。

2.3 算法分析

本文提出的 HCSDP 算法, 实质属于一种范围划分策略, 划分的边界为 V_{avg} 值。由于 Hilbert 空间填充曲线具有良好的空间聚类特性, 使得空间上相邻的对象可以划分到同一个磁盘或相邻的磁盘上去, 进而保证在范围查询时访问的磁盘数

量尽量少。因为每个磁盘所存储的空间实体大小相对均衡, 每个磁盘用于存储空间实体对象的页数相差不多, 这样不但可以平衡数据访问负载, 同时可以使系统获得较大的数据吞吐能力和数据的并发处理能力。表3和表4是 HCSDP 算法同图1所示算法在范围查询实验中的响应时间对照表。表中的查询边界是指查询范围框的边线长, 这个长度值是经过归一化处理后的取值, 即将空间对象坐标值映射到 $R = [0, 1]^2$ 的空间上。表中每一个查询边界值对应 100 次的查询, 其中的查询起始点随机选取, 而查询响应时间为 100 次的平均时间, 单位为 ms。

表3 3个磁盘下范围查询对照表

Tab. 3 Result of Range Query Based on 3 Disks

查询边界	HCSDP 算法	Oracle Spatial 算法
0.1	221	238
0.2	706	897
0.3	1 487	1 863

表4 5个磁盘下范围查询对照表

Tab. 4 Result of Range Query Based on 5 Disks

查询边界	HCSDP 算法	Oracle Spatial 算法
0.1	202	215
0.2	598	832
0.3	1 237	1 634

从表3和表4的结果对比中不难发现, 当查询范围扩大时, 两种算法所对应的系统响应时间差别较大; 而查询范围较小时, 其响应时间差别较小。总体来说, HCSDP 算法所对应的系统响应时间少于采用 Oracle Spatial 的数据划分算法的系统响应时间。其主要原因是: HCSDP 算法使空间邻近的地物实体位于同一个或相邻的磁盘上, 而 Oracle Spatial 中采用的策略则不同。当查询范围扩大时, 前者所需要访问的磁盘数目要少于后者, 从而节约了访问多个磁盘的响应时间; 而查询范围较小时, 两者相差较小, 甚至会出现个别查询速度后者快于前者的情况。例如, 当查询范围恰好落在图1所示的某个格网单元之内时, 查询范围不需要跨多个格网单元, 只需要访问一个磁盘。然而, HCSDP 算法需要计算查询范围所对应的 Hilbert 值域, 这将带来一部分时间开销。

3 结语

并行空间数据库是并行 GIS 应用的基础, 同时也将为未来网格计算在 GIS 中的应用提供强大的数据服务功能。空间数据划分方法和策略是

并行空间数据库的关键,其关系到并行空间数据库的整体性能和服务能力。本文的 HCSDP 算法为并行空间数据库中空间数据的划分提供了一种新的策略。实践证明,该方法可获得平衡的负载以及较高的空间数据并行查询效率。

参 考 文 献

- [1] Michael J M, Steve D, Bruce M G. Towards a HPC Framework for Integrated Processing of Geographical Data: Encapsulating the Complexity of Parallel Algorithms [J]. *Trans in GIS*, 2000, 4(3): 245-262
- [2] Dowers S, Gittings B M, Mineter M J. Towards a Framework for High-performance Geocomputation: Handling Vector-topology Within a Distributed Service Environment [J]. *Computers, Environment and Urban Systems*, 2000(24): 471-486
- [3] Michael J M. A Software Framework to Create Vector-topology in Parallel GIS Operations[J]. *International Journal of Geographical Information Science*, 2003, 17(3): 203-222
- [4] Richard H, Steve D, Bruce T, et al. *Parallel Processing Algorithms for GIS* [M]. UK: Taylor & Francis Ltd, 1998
- [5] Nickerson B G, Gao F. Spatial Indexing of Large Volume Swath Data Sets[J]. *International Journal of Geographical Information Science*, 1998, 12(6): 537-559
- [6] Jonathan L. *The Application of Space-filling Curves to the Storage and Retrieval of Multi-dimensional Data* [D]. London: University of London, 2000
- [7] Kumar A, M uhanha W A. Analysis of the Performance of Spatial Ordering Methods[J]. *International Journal of Geographical Information Science*, 1998, 12(3): 269-289
- [8] 陆锋, 周成虎. 一种基于 Hilbert 排列码的 GIS 空间索引方法[J]. *计算机辅助设计与图形学学报*, 2001, 13(5): 424-429
- [9] Bongki M, Jagalish H V, Christos F, et al. Analysis of the Clustering Properties of the Hilbert Space Filling Curve [J]. *IEEE Trans on Knowledge and Data Engineering*, 2001, 13(1): 124-141

第一作者简介: 赵春宇, 博士, 主要研究方向为 Internet GIS、并行计算及高性能并行 GIS 理论与技术。

E-mail: zcy7@263.net

Spatial Data Partitioning Towards Parallel Spatial Database System

ZHAO Chunyu¹ MENG Lingkui¹ LIN Zhiyong¹

(¹ School of Remote Sensing and Information Engineering, Wuhan University, 129 Luoyu Road, Wuhan 430079, China)

Abstract: A new algorithm of spatial data partitioning is presented based on Hilbert curve which is one of the space filling curves. The new spatial data partitioning HCSDP algorithm gives a new vector data partitioning policy on the parallel spatial database system. Taking the characteristic of the variant-length of vector data records into account, HCSDP algorithm divides uniformly the huge volume of spatial data set into multiple parts and putting them onto the different processing nodes for avoiding the data skew. The advantages of HCSDP comparing are discussed with the algorithm used in Oracle spatial, and the validity and the efficiency of the algorithm are illustrated by the experimental results.

Key words: parallel spatial database; data partitioning; data skew; Hilbert space filling curve

About the first author: ZHAO Chunyu, Ph. D. His research interests are Internet GIS, parallel computing and the theory and techniques of high performance parallel GIS.

E-mail: zcy7@263.net