

DOI:10.13203/j.whugis20160271



文章编号:1671-8860(2018)07-1092-06

云环境下海量空间矢量数据并行划分算法

姚晓闯^{1,2,3} 杨建宇^{1,2} 李林^{1,2} 叶思菁³ 郎文聚^{1,2,4} 朱德海^{1,2}

1 中国农业大学信息与电气工程学院,北京,100083

2 国土资源部农用地质量与监控重点实验室,北京,100035

3 中国科学院遥感与数字地球研究所,北京,100094

4 国土资源部土地整治中心,北京,100035

摘要:空间数据划分是空间大数据索引方法及其数据存储的重要组成部分。针对 Hadoop 云计算平台在空间数据划分及其存储方面的不足,提出了基于 Hilbert 空间填充曲线的海量空间矢量数据并行划分算法。在数据划分阶段,充分考虑空间数据相邻对象的空间位置关系、空间对象的自身大小以及相同编码块的空间对象个数等影响因素;通过“合并小编码块,分解大编码块”的划分原则,实现了云环境下海量空间矢量数据的并行划分算法。试验表明,该算法不仅能够提高海量空间矢量数据的索引效率,同时也能够很好地解决空间矢量数据在 Hadoop 分布式文件系统(Hadoop distributed file system, HDFS)上的数据倾斜问题。

关键词:矢量数据;Hilbert 编码;空间数据划分;MapReduce;R-tree 索引;数据倾斜

中图分类号:P208

文献标志码:A

空间数据划分的目的是将空间大数据分割为相对较小的、独立的多数据块进行存储,能够很大程度上提高空间数据的分布式存储和并行化处理能力。在众多传统的空间数据划分算法中,由于 Hilbert 空间填充曲线具有良好的空间集聚特征^[1-3],因此被广泛应用于空间数据划分领域。文献[4]提出了一种基于 Hilbert 空间填充曲线的矢量数据划分算法,该算法按照 Hilbert 编码排序,通过遍历所有要素,依次写入 N 个数据桶中,并映射到 N 个存储节点上。文献[5]通过 K-均值聚类算法得到 Hilbert 编码对应的质心,再根据数据量大小进行空间划分并存储在 K 个节点上。文献[6]提出了基于 Hilbert 曲线层次分解的空间数据划分方法,对大数据块进行多级分解,然而当数据块过大时,势必会导致层次分解过多,从而降低数据划分效率。传统的空间数据划分方法尽管能够很好地保证数据划分结果的空间分布聚集特征,但当数据量过大时,单节点的数据划分算法很容易成为限制瓶颈;同时,在存储节点上,数据仍然以整体数据块存在,随着数据量的增加,同样会影响数据的检索效率。

云计算技术的出现为大数据的存储与管理提供了理想的解决方案^[7-9]。其中,Hadoop 平台具有易扩展、高容错、可靠、高效、经济等特点,近几年在科学计算、人工智能等多个领域都得到了广泛的应用。然而在空间大数据处理方面,由于空间数据独有的空间特征,使得 Hadoop 平台并不能充分发挥自己的优势^[10]。基于 Hadoop 默认的 Hash 划分方法,尽管能够很好地保证数据块之间的均衡,但往往会导致同一空间要素或相邻空间要素存储在不同的数据块中,这样会大大降低空间数据的索引效率以及空间操作的执行性能。

为了弥补 Hadoop 平台中数据划分及其存储的不足,文献[11-12]通过抽取随机样本,建立了相应的空间数据划分策略,实现了格网、四叉树、Hilbert 等 7 种不同的空间数据划分并行化算法。但由于样本的随机性,对于空间索引而言,一方面无法保证其空间索引结果的一致性;另一方面也会丢失数据的空间分布特征,从而导致空间数据的划分结果并不理想,影响空间索引效率。对于 Hadoop 平台而言,由于空间数据的分布不均,很

收稿日期:2017-02-05

项目资助:国土资源部公益性行业科研专项基金(201511010-06)。

第一作者:姚晓闯,博士,主要从事空间大数据研究。xcyao@cau.edu.cn

通讯作者:朱德海,博士,教授。dehaihu@qq.com

容易造成个别 Reduce 任务负载“过热”,使得整个任务执行效率降低,其结果也将直接导致 Hadoop 分布式文件系统(Hadoop distributed file system, HDFS)中的数据块分布严重倾斜。

本文针对以上技术的不足,提出了一种云环境下的海量空间矢量数据并行划分算法,详细介绍了算法的实现过程,并从 R-tree 空间索引效率和 HDFS 存储数据倾斜度两个方面比较了本文算法的优越性。

1 Hilbert 空间编码

Hilbert 曲线是经典的 Peano 空间填充曲线族中的一种,由德国数学家希尔伯特于 1981 年构造出该曲线空间填充的几何过程^[13]。Hilbert 曲线定义为 N 维空间 $R(N)$ 与一维空间 R 之间的一一映射。如图 1 所示,分别为 1 阶和 2 阶 Hilbert 空间填充曲线及其对应的空间编码。

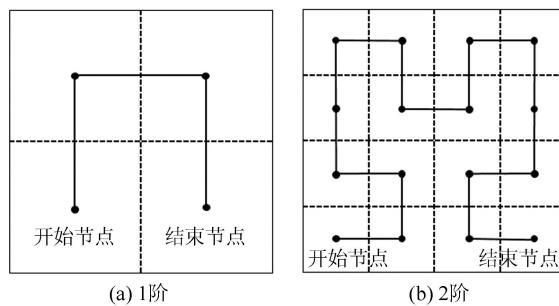


图 1 1 阶和 2 阶 Hilbert 编码

Fig.1 1st and 2nd-orders Hilbert Codes

常见的空间填充曲线还有 Z 曲线、Gray 曲线等,但由于 Hilbert 空间填充曲线在生成过程中没有出现空间上的大幅度跳跃,使得编码在空间分布上具有良好的聚集性,即相邻 Hilbert 空间编码的编码块在空间分布上一定是相邻的。因此,本文算法在数据划分过程中,充分利用 Hilbert 曲线这一优点,使得该算法能够在很大程度上保证空间数据的分布特征,从而提高空间索引效率。

2 空间矢量数据划分算法描述

本文算法主要针对空间矢量数据,即点、线、面数据要素。其中,对于点要素,直接用坐标点来表示空间位置信息;对于线和面要素,用其对应的中心点坐标来代替。假设 Hilbert 空间填充曲线的阶数为 N ,则整个数据集 D 的空间范围将被划

分为 $2^N \times 2^N$ 个格网。每一个格网具有唯一的 Hilbert 编码标识,将该格网称之为编码块。

在该算法中,Hilbert 编码阶数的初始值设定为 N_0 :

$$N_0 = \lceil \log_2 \frac{V}{V'} \rceil \quad (1)$$

式中, V 为数据量总体大小; V' 为 HDFS 上默认的数据块存储大小; $\lceil \cdot \rceil$ 表示向上取整。具体算法描述为:

1) Hilbert 空间编码。通过遍历要素集,获取每一个空间要素在 N 阶格网下对应的 Hilbert 空间编码、中心点坐标和数据量大小。

2) 空间信息统计。统计具有相同编码(H_{code})的要素对应的信息集 S ,即每一个编码块对应的数据量大小(H_{size})和数据量个数(H_{count})。如果该编码块数据量大小(H_{size})大于 HDFS 数据块存储大小($V' \times \rho_{max}$),则记录该编码块对应的二级划分样本集合 s ;否则, $s = \{0\}$ 。其中 ρ_{max} 为 HDFS 存储数据块的最大阈值百分比。

3) 生成空间划分矩阵 T 。根据步骤 2) 中得到的空间数据样本信息集 S ,基于“合并小编码块,分解大编码块”的划分思想,利用 Hilbert 空间填充曲线的聚集特性,将相邻的小编码块进行合并;同时,根据二级划分集合 s ,将大编码块进行再划分,最终可以计算得出每一个编码块(H_{code})对应的 HDFS 数据块存储编号(i),从而生成该数据集对应的空间数据划分矩阵 T :

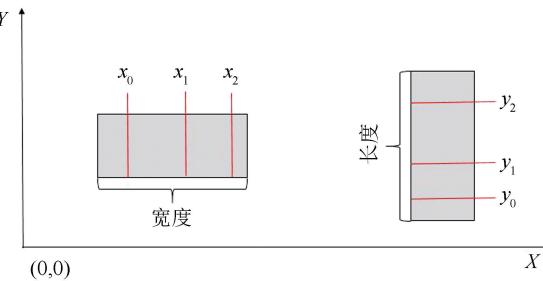
$$T = \begin{bmatrix} H_{code_0} & i_0 & s_0 \\ H_{code_1} & i_1 & s_1 \\ \vdots & \vdots & \vdots \\ H_{code_n} & i_n & s_n \end{bmatrix} \quad (2)$$

4) 空间数据划分。根据空间数据划分矩阵 T ,通过再次遍历空间要素集,获取其对应的 Hilbert 空间编码和矩阵 T 匹配对应的 HDFS 数据块存储编号,并将该要素写入到该数据块中,至此完成所有空间要素的划分工作。

为了使得基于该划分方法的空间索引效率更优,在步骤 2) 中,如图 2 所示,通过单向计算其对应的二级划分集合 s 。如果宽度大于长度,计算 X 方向集合 $\{x_0, x_1, x_2 \dots x_t\}$;否则,计算 Y 方向的集合 $\{y_0, y_1, y_2 \dots y_t\}$ 。利用式(3)获取元素间隔大小:

$$L = \frac{V'}{\bar{V}} \quad (3)$$

式中, L 代表间隔大小; \bar{V} 代表该编码块要素的平均大小。

图 2 二级划分集合 s 的获取Fig.2 Acquisition of the Sub-split Set s

通过该方法进行大编码块的再次分解,与文献[6]相比,大编码块有且只有一次被执行再处理,有效降低了算法的复杂度。同时对于分割后

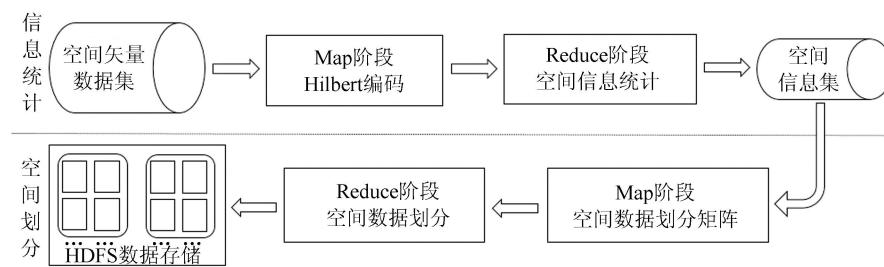


图 3 算法并行化实现
Fig.3 Implementation of the Parallel Algorithm

3.1 空间信息统计任务

Map 阶段:在 Map 过程中,读入的值是单个空间要素,通过 Map 函数,实现空间要素的键值化。其中键为 Hilbert 空间编码(H_{code}),值为空间要素字节大小(E_{size})和中心点坐标(C_{point}),则 Map 输出的一条记录可表述为 $\langle H_{code}, E_{size}; C_{point} \rangle$ 。

Reduce 阶段:在 Reduce 过程中,读入的键为编码块对应的 Hilbert 空间编码(H_{code}),而值为该编码块对应的所有要素信息集合,即 $\langle H_{code}, (E_{size}; C_{point}) (E_{size}; C_{point}) \dots (E_{size}; C_{point}) \rangle$ 。通过 Reduce 函数,首先利用式(3)计算该编码块大小(H_{size});然后与 HDFS 中默认的存储数据块大小($V' \times \rho_{max}$)进行比较,判断是否进行二级划分,并记录其二级划分集合 s 。Reduce 输出结果的键保持不变,值为 $\langle H_{size}; s \rangle$,每一条记录的键值对为 $\langle H_{code}, H_{size}; s \rangle$ 。二级划分集合 s 的计算方法见图 2。 H_{size} 的计算方法为:

$$H_{size} = \sum_{i=0}^t E_{size}(i) \quad (4)$$

至此,空间信息统计阶段结束,那么由该阶段

的碎片数据依旧判断是否与相邻数据块进行合并。

在步骤 3)中,依据 Hilbert 曲线空间聚集特性,将相邻的小编码块进行合并,在空间划分矩阵 T 中,它们会对应相同的 HDFS 数据块存储编号,同时在步骤 4)中会被写入到同一个数据块中。

3 空间矢量数据并行划分算法实现

本文算法在具体实现过程中通过两个 MapReduce 任务来完成,如图 3 所示,包括空间信息统计阶段和空间数据划分阶段。

生成的空间信息集 S 为所有记录的集合, S 可表示为:

$$S = \{ \langle H_{code_0}, H_{size}; s \rangle, \langle H_{code_1}, H_{size}; s \rangle, \dots, \langle H_{code_t}, H_{size}; s \rangle, \dots, \langle H_{code_t}, H_{size}; s \rangle \} \quad (i \in [0, t])$$

一般情况下,空间矢量要素分布是不均匀的,所以 t 小于 Hilbert 曲线在 N 阶下编码块的总个数 2^{2N} 。

3.2 空间数据划分任务

Map 阶段:在 §2 的步骤 3)中,针对空间信息集 S ,生成空间数据划分矩阵 T 。在 Map 阶段,通过比较编码块大小(H_{size})与 HDFS 存储数据块大小($V' \times \rho_{max}$),合并或分解编码块,并计算当前编码块在 HDFS 上的存储数据块编号(i)。

Reduce 阶段:主要实现空间数据的具体划分和数据块的分发工作。通过获取当前空间要素对应的 H_{code} ,与空间数据划分矩阵 T 进行匹配,如果二级划分样本集合 $s = \{0\}$,则直接写入到对应的编号为 i 的数据块中;否则,计算其 x 值或者 y 值在集合 s 中的排序位置 i' ,将其写入到编号为 $(i + i')$ 的数据块中。通过遍历所有空间要素集,完成空间数据在 HDFS 上的数据库划分工作。

4 试验与结果分析

4.1 对比试验设计

试验环境采用 Hadoop 集群,节点个数为 4,操作系统为 CentOS 6,Hadoop 版本为 Hadoop-1.2.1,HDFS 存储数据块默认大小为 64 MB。其中每个节点的内存大小为 16 GB,磁盘容量为 1 TB。

实验数据分别采用全球县级行政区划数据(D_0)、全球湖泊分布数据(D_1)和全球矢量数据集(D_2)。数据基本信息如表 1 所示。

表 1 实验数据集信息

Tab.1 Information of Experimental Datasets

数据名称	数据大小/GB	要素个数/MB	平均大小/kB
D_0	2.9	0.255	11.9
D_1	9.1	8.4	1.14
D_2	92.5	263	0.369

试验采用两种方案,一种为文献[11]基于随机抽样的 Hilbert 数据划分方法,另一种为本文所提出的划分方法。在随机抽样中,设定抽样率为 1%^[11]。本文方法中,Hilbert 的阶数 N 设定为 10。

4.2 试验结果分析

本文通过两个方面对试验结果进行分析,分别为 R-tree 空间索引性能和 HDFS 存储数据倾斜度。

在 R-tree 空间索引性能方面,通过对两种划分方法分别建立对应的 R-tree 空间索引^[10],对比 R-tree 索引的性能指标 Area(T)和 Overlap(T)。其中 Area(T)为 R-tree 索引树中闲置区域的覆盖总面积,闲置区域越少,则 R 树索引性能越好;Overlap(T)为 R-tree 索引树中重叠区域的覆盖总面积,重叠区域越少,其索引性能越好。标准化后的结果分别见图 4、图 5。

从图 4 和图 5 中可以看出,本文提出的方法在 Area(T)和 Overlap(T)两方面都表现优越。基于文献[11]中的方法,由于样本的随机性而丢失了数据划分的空间特征。本文方法充分利用了 Hilbert 空间填充曲线的空间相邻特征,基于 Hilbert 编码进行数据划分,很大程度上保证了矢量数据的空间分布特征,因而能够很好地提高空间数据的索引效率。

在 HDFS 存储数据倾斜度方面,计算两种划

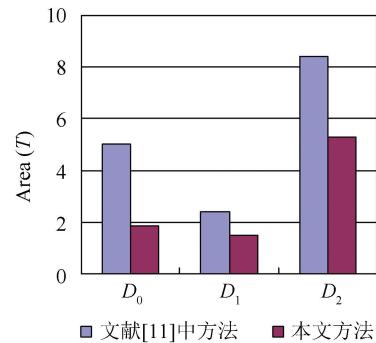


图 4 不同数据集的性能指标 Area(T)对比

Fig.4 Comparison of Performance Index Area(T) for Different Datasets

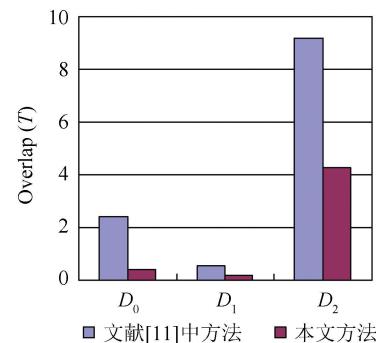


图 5 不同数据集的性能指标 Overlap(T)对比

Fig.5 Comparison of Performance Index Overlap(T) for Different Datasets

分方法对应的所有数据块大小的标准差,如图 6 所示。标准差越小,表明数据倾斜度越小,数据在 HDFS 上的分布就越均匀。由图 6 可知,本文方法在 HDFS 数据倾斜方面表现较优。由于本文充分考虑了空间数据对象的自身大小和空间编码块的个数,并通过与 HDFS 默认的存储数据块进行对比,采用“合并小编码块,分解大编码块”的思想,在不丢失空间分布特征的前提下,很好地保证了 HDFS 数据的负载均衡。

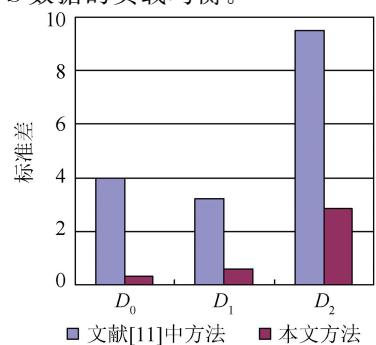


图 6 不同数据集的 HDFS 数据倾斜对比

Fig.6 Comparison of Data Skewness in HDFS for Different Datasets

5 结语

本文提出了一种基于 Hadoop 平台的海量空间矢量数据并行划分算法。一方面,将 Hilbert 空间填充曲线引入到数据划分规则中,使得同一存储数据块中的要素在空间上保持相邻,提高了空间索引效率;另一方面,将空间对象的自身大小、相同编码块的空间对象个数、HDFS 的存储块大小等影响因素设为参考,很好地解决了空间矢量数据在 HDFS 中分布不均的问题。试验表明,该方法在空间索引效率和 HDFS 存储数据倾斜度方面均具有优越性,能够为空间矢量大数据的分析与处理提供更好的数据服务。

参 考 文 献

- [1] Lu Feng, Zhou Chenghu. An Algorithm for Hilbert Ordering Code Based on Spatial Hierarchical Decomposition[J]. *Journal of Image and Graphics*, 2001, 6(5): 465-469(陆锋,周成虎. 一种基于空间层次分解的 Hilbert 码生成算法[J]. 中国图象图形学报, 2001, 6(5): 465-469)
- [2] Guo Jing, Liu Guangjun, Dong Xurong, et al. 2-Level R-tree Spatial Index Based on Spatial Grids and Hilbert R-tree[J]. *Geomatics and Information Science of Wuhan University*, 2005, 30(12): 1 084-1 088(郭晶,刘广军,董绪荣,等. 基于空间网格和 Hilbert R-tree 的二级 R-tree 空间索引[J]. 武汉大学学报·信息科学版, 2005, 30(12): 1 084-1 088)
- [3] Dai Jing, Wu Mingguang, Zheng Peibei, et al. An Improved STR-tree Spatial Index Algorithm Based on Hilbert-curve[J]. *Geomatics and Information Science of Wuhan University*, 2014, 39(7): 777-781(戴晶,吴明光,郑培蓓,等. 基于 Hilbert 曲线的 STR 索引改进算法[J]. 武汉大学学报·信息科学版, 2014, 39(7): 777-781)
- [4] Zhao Chunyu, Meng Lingkui, Lin Zhiyong. Spatial Data Partitioning Towards Parallel Spatial Database System[J]. *Geomatics and Information Science of Wuhan University*, 2006, 31(11): 962-965(赵春宇,孟令奎,林志勇. 一种面向并行空间数据库的数据划分算法研究[J]. 武汉大学学报·信息科学版, 2006, 31(11): 962-965)
- [5] Wang Yongjie, Meng Lingkui, Zhao Chunyu. Spatial Partitioning of Massive Data Based on Hilbert Spatial Ordering Code[J]. *Geomatics and Information Science of Wuhan University*, 2007, 32(7): 650-653(王永杰,孟令奎,赵春宇. 基于 Hilbert 空间排列码的海量空间数据划分算法研究[J]. 武汉大学学报·信息科学版, 2007, 32(7): 650-653)
- [6] Zhou Yan, Zhu Qing, Zhang Yeting. A Spatial Data Partitioning Algorithm Based on Spatial Hierarchical Decomposition Method of Hilbert Space-Filling Curve[J]. *Geography and Geo-Information Science*, 2007, 23(4): 13-17(周艳,朱庆,张叶廷. 基于 Hilbert 曲线层次分解的空间数据划分方法[J]. 地理与地理信息科学, 2007, 23(4): 13-17)
- [7] Xiong Lian, Xu Zhengquan, Wang Tao, et al. On the Store Strategy of Small Spatio-Temporal Data Files in Cloud Environment[J]. *Geomatics and Information Science of Wuhan University*, 2014, 39(10): 1252-1256(熊炼,徐正全,王涛,等. 云环境下的时空数据小文件存储策略[J]. 武汉大学学报·信息科学版, 2014, 39(10): 1 252-1 256)
- [8] Zhang Xiaoxiang. Spatial Analysis in the Era of Big Data [J]. *Geomatics and Information Science of Wuhan University*, 2014, 39(6): 655-659(张晓祥. 大数据时代的空间分析[J]. 武汉大学学报·信息科学版, 2014, 39(6): 655-659)
- [9] Xu Wen, Shao Jun, Yu Wenyong, et al. Land Observing Satellite Data Center: Big Data Challenges and a Potential Solution[J]. *Geomatics and Information Science of Wuhan University*, 2017, 42(1): 7-13(徐文,邵俊,喻文勇,等. 陆地观测卫星数据中心:大数据挑战及一种解决方案[J]. 武汉大学学报·信息科学版, 2017, 42(1): 7-13)
- [10] Eldawy A, Mokbel M F. Spatial Hadoop: A Map Reduce Framework for Spatial Data[C]. IEEE 31st International Conference on Data Engineering, Seoul, Korea, 2015
- [11] Eldawy A, Alarabi L, Mokbel M F. Spatial Partitioning Techniques in Spatial Hadoop[C]. The International Conference on Very Large Databases, VLDB 2015, Kohala Coast, Hawaii, 2015
- [12] Li Xun. Parallel Spatial Index Algorithm Based on Hilbert Partition[D]. Chengdu: University of Electronic Science and Technology of China, 2013(李勋. 基于 Hilbert 划分的并行矢量数据索引算法研究[D]. 成都:电子科技大学, 2013)
- [13] He Xiaoyuan, Min Huaqing. Hilbert R-tree Spatial Index Algorithm Based on Clustering[J]. *Computer Engineering*, 2009, 35(9): 40-42(何小苑,闵华清. 基于聚类的 Hilbert R-树空间索引算法[J]. 计算机工程, 2009, 35(9): 40-42)

Parallel Algorithm for Partitioning Massive Spatial Vector Data in Cloud Environment

YAO Xiaochuang^{1,2,3} YANG Jianyu^{1,2} LI Lin^{1,2} YE Sijing³ YUN Wenju^{1,2,4} ZHU Dehai^{1,2}

1 College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China

2 Key Laboratory for Agriculture Land Quality, Monitoring and Control of the Ministry of Land and Resources, Beijing 100035, China

3 Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100094, China

4 Land Consolidation and Rehabilitation Center, Ministry of Land and Resources, Beijing 100035, China

Abstract: Spatial data partitioning plays an important role in the spatial index methods and the data storage strategy for spatial big data. In this paper, to make up the inherent shortcomings of spatial data partitioning and data storage in the Hadoop cloud computing platform, a parallel algorithm based on Hilbert space-filling curve is presented for partitioning the massive spatial vector data. In the spatial vector data partitioning phase, we take more influence factors, including the spatial location relationship between adjacent objects, the size of spatial vector object itself, the number of spatial objects in the same spatial coded block and others, into full consideration. Meanwhile, by following the partitioning principle of merging small coded blocks and sub-splitting large coded blocks, this paper implements the parallel algorithm for partitioning the massive spatial vector data in cloud environment. Experimental results show that the algorithm proposed in this paper can not only improve the efficiency of the spatial R-tree index for massive spatial vector data, but also give a good data balance in Hadoop distributed file system (HDFS).

Key words: vector data; Hilbert code; spatial data partitioning; MapReduce; R-tree index; data skewness

First author: YAO Xiaochuang, PhD, specializes in spatial big data. E-mail: xcyao@cau.edu.cn

Corresponding author: ZHU Dehai, PhD, professor. E-mail: dehaizhu@qq.com

Foundation support: The Special Fund for Scientific Research of Public Welfare Industry of Ministry of Land and Resources, No. 201511010-06.