一种基于动态散列的 GIS 空间索引构造算法

陈文生　米　红　张希雯
(厦门大学信息科学与技术学院模式识别与智能系统研究所,厦门 361005)
E-mail: Mihong2017@vip.sina.com

摘　要　文章在介绍动态散列和传统空间索引四叉树的构造方法的基础上,综合二者的优点,提出了一种基于动态散列的空间索引构造算法,该方法改变了传统四叉树通过效率低下的空间对象的递归比较构造索引过程,采用计算机运算效率较高的二进制位运算和位比较的动态散列扩充散列值来构造空间索引。实践证明,该算法大大减少了空间索引的构造时间和效率,具有很高的应用价值。

关键词　动态散列　空间索引　四叉树　GIS

文章编号 1002-8331-(2006)08-0173-02　**文献标识码** A　**中图分类号** TP301.6

Algorithm of Spatial Query in GIS Based on Dynamic Hash

Chen Wensheng　Mi Hong　Zhang Xiwen
(College of Science and Technology, Xiamen University, Xiamen 361005)

Abstract: Introducing dynamic hash function and original quadtree, this paper assigns an algorithm of spatial query based on dynamic hash function.Instead of spatial object's recursive comparison where original quadtree has been used, the algorithm builds the spatial index by applying binary code operation in which computer runs more efficiently, and extended dynamic hash code for bit comparison.Experiment results show that the algorithm is efficient, simple and has powerful practical merits.

Keywords: dynamic hash, spatial query, quadtree, GIS

1　引言

构造索引是提高查询效率的一种非常重要的方法。动态散列在构造关系数据库索引方面应用很普遍,但在构造空间索引方面由于很难找到一个可以从把空间物体影射到散列值的哈希函数而应用不多。四叉树作为一种重要的空间索引构造技术由于其高效的数据压缩,操作简单,实现效率高,常常被用来构造二维空间数据的索引，但是构造四叉树构成是一个递归过程,空间实体对象在四叉树的定位是从顶层结点开始一级一级地往下比较,直到找到合适的位置,才插到四叉树上的,而且这种比较是空间对象跟空间对象的位置比较,这种比较只能通过计算机的高级语言来实现,而动态散列法是逐步扩充散列值的位数来来构造索引,更主要的是它是通过位比较来实现散列值的定位，这种比较方式计算机通过几个 CPU 机器指令即可实现,故它的效率相对于四叉树的空间对象跟空间对象的比较而言具有更高的效率。故本文基于动态散列技术提出了一种高效的空间索引的构造方法。

2　动态散列概述

散列又叫哈希,它是根据哈希函数和冲突处理的方法将一组关键字影射到一个有限的连续地址集上,并以关键字在地址集中的"象"作为存储位置。跟一般的查找方法(如线性表、树等)相比,它在定位过程中不用进行关键字的比较,查找效率不依赖查找过程所进行的比较次数,记录存储位置和关键值存在着一个确定的对应关系，通过关键值就可以影射到其存储地址,因此具有较高的查找效率。

动态散列技术允许散列函数动态改变,通过桶的合并和分解来实现数据库的增大或缩小的需求,这样既继承了了散列的高效查找效率,又保持了良好的空间压缩率。其构造过程如下:

首先,根据哈希函数影射出各记录的一组具有 b 位二进制数的散列值。

其次,根据逐步扩充使用二进制数散列值的位数构造散列表,实现的建立。

开始时,使用散列位数为零,初始化一个空桶,记录逐条地插入其中,当桶已满时,再往里添加会引起使用散列位数的增加和桶的分裂，原桶里的记录按照新的散列值添加到新桶里,当前使用散列位数加一。

表 1 是待建立索引的文件,表 2 是根据索引码构造的哈希表,图 1 是根据动态哈希建立的散列结构:这里假设桶的容量为 2，我们可以看出虽然散列值的位数为 4，但只用两位就可以,而且散列值的比较可以是位运算,这相对于一般四叉树构造过程中空间对象与空间对象的位置分析效率要高出许多(具体构造过程参见参考文献[2])。

3　基于动态散列技术构造空间索引

空间索引是空间数据库的关键技术,其性能的高低决定了整个空间数据库的效率。常用空间索引有 R 树类空间索引和

基金项目:福建省自然科学基金资助项目(编号:A0410006);厦门大学 985"海量数据挖掘"研究项目资助

作者简介:陈文生,厦门大学信息科学与技术学院模式识别与智能系统研究所硕士研究生。米红(1962-),男,系统工程专业博士,GIS 研究方向博士后,厦门大学信息科学与技术学院模式识别与智能系统研究所所长,厦门大学人口资源环境与地理信息系统研究中心主任。张希雯,硕士研究生。

表 1　Account 例子文件

字段 1(索引码)	字段 2	字段 3
Brighton	A_217	…
Downtown	A_753	…
perridge	A_253	…
Redwood	A_143	…
Round	A_453	…

表 2　散列函数表

关键值	散列值
Brighton	0010
Downtown	1010
Perridge	1111
Redwood	0011
Round	1101

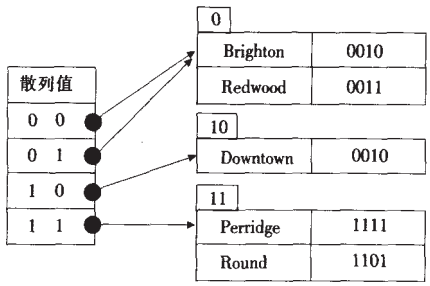

图 1　Account 文件的散列结构

四叉树类空间索引。目前国内外主要的空间数据库大都采用这两类空间索引方法。国外 Esri 的 ArcView、Mapinfo 公司的 Mapinfo 和 Informix 的 GeoSpatial DataBlade 采用的是 R 树系列作为空间索引；国内地质大学的 MapGis 和中科院的 SuperMap 则采用的是四叉树。而著名的 ORACLE 公司的 Spatialware 则同时采用这两类索引方法。

四叉树基于空间划分组织索引结构的一类索引机制。将已知范围的空间划成四个相等的子空间，如果需要可以将每个或其中几个子空间继续划分下去，这样就形成了一个基于四叉树的空间划分。由此可见构造四叉树索引的过程是一个递归过程，空间对象在四叉树上定位是从根结点开始“自上而下”一一比较，找到叶结点才确定其插入位置，这种比较是通过空间对象和空间对象的比较，在计算机中要通过高级语言来实现，因此构造四叉树的时间相对长。

基于动态散列技术构造四叉树索引是采用“自下而上”的构造方法。这种构造过程是通过散列值的动态扩充和二进制数的位比较，计算机通过简单的几个机器指令即可实现，因此效率很高，其基本思想是：根据用户情况把地图划分成 $2^n \times 2^n$（n 为大于 0 的自然数）的单元格，根据每个单元格的空间信息影射出一个 2*n 位二进制数，这个过程相当于动态散列法的散列过程，其二进制数相当于散列值，再根据动态散列的方法构造空间索引。

下面通过动态散列技术对一组空间对象构造空间索引。

图 2 是一幅既有 9 个空间对象的地图及其四叉树构造情况。这里把地图划分成 $2^2 \times 2^2$ 的单元格，每个桶最多能放 2 个空间对象。动态散列技术构造空间索引重要的一步是影射单元格的散列值。把地图划分成 $2^n \times 2^n$ 份是通过 X 轴划分 2^n 份，Y 轴

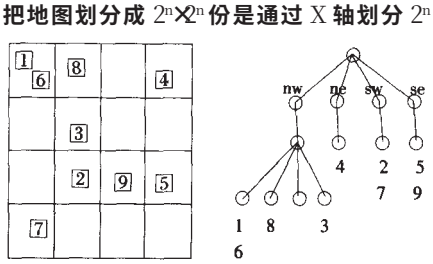

图 2　对象空间分布及在四叉树上的结点分配

划分 2^n 份而形成的，我们可以用 $0 \sim 2^n - 1$ 来标识单元格在 X、Y 轴上的位置，这个位置值用一个 n 位的二进制表示，则单元格的散列值为 $X_1Y_1 \cdots X_{n-1}Y_{n-1}X_nY_n$ 如某一单元格的 X、Y 轴坐标值分别为 01、10，则其散列值为 0110。

图 3 是对整个地图空间网格影射的散列值列表，图 4 是根据空间对象所处的网格影射出的散列值。

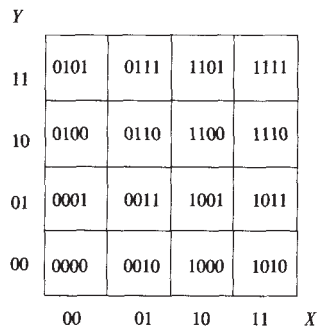

图 3　空间网格及其散列值

散列值	对象
0101	空间对象 1
0011	空间对象 2
0110	空间对象 3
1111	空间对象 4
1011	空间对象 5
0101	空间对象 6
0000	空间对象 7
0111	空间对象 8
1001	空间对象 9

图 4　空间对象散列值

影射出空间对象的散列值后，就可根据动态散列法构造其散列结构（这里假设桶的长度为 2）：图 5 是使用 0 位散列值，桶的长度为 2，可插入 2 个空间对象。插入对象 3 时桶已经装满，则分裂桶，同时使用散列值的位数加 1，但对象 1、2、3 的第一位散列值均为 0，所以还必须进行分裂桶和增加使用散列值的位数，图 6 是插入对象 3 后的散列结构。图 7 是空间对象的最终散列结构。

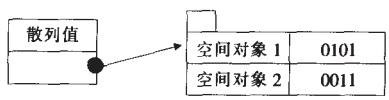

图 5　使用散列值位数为零

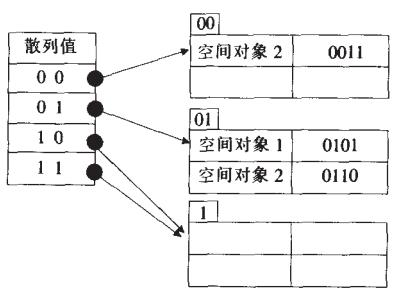

图 6　使用散列值位数为 2

基于动态散列技术构造空间索引的优点见表 3。表 4 的数据是通过 ESRI 的 MapObject 组件和 Shape 文件实现空间对象坐标的读取，因此包含 Com 组件的交互时间。但是我们还是可以看到动态散列法构造空间索引的时间相对与四叉树还是要少的多，动态散列法的效率比四叉树要高的多。

4　结束语

空间索引是空间数据库的重要技术，设计高效的索引能极大地提高整个空间数据库的效率。本文在分析常见空间索引四叉树的缺陷基础上，提出了一种基于动态散列技术适合于计算

（下转 189 页）

(3)根据 L[2]得到其等价类集合:

$L[2]/R=\{E_{2,1}, E_{2,2}\}$

$E_{2,1}=\{I1I2, I1I3, I1I5\}$

$E_{2,2}=\{I2I3, I2I4, I2I5\}$

$2[I1]=\{I2, I3, I5\}$

$E_2[I2]=\{I3, I4, I5\}$

连接 S 与每个等价类中的元素得到:

$E_3[I1I2]=E_2[I1]\cap E_2[I2]=\{I3, I5\}$;

依次类推,可以得到所有的 E_3。

由于其它的 $E_3[S]$都为空,所以连接 $E_3[I1I2]$中各元素和 S=I1I2,可以得到 C[3]={{I1, I2, I3}, {I1, I2, I5}}。在依据(2)最终得到 L[3]={{I1, I2, I3}, {I1, I2, I5}}。

通过示例可以知道,利用等价类和等价关系挖掘关联规则算法时,在对 D 的扫描次数及侯选频繁项集上均占有较大优势。

为了进一步验证算法的性能,我们在运行 Windows 2000 操作系统的 PentiumIII-800MHz(内存 128MB)计算机上做了测试,使用了人工合成的数据[3],合成以后数据的各项参数如下:数据记录集中记录的个数为 D=5 000~25 000;各条记录中包含项目的平均数 T=5,项目的个数 N=20;横坐标表示支持度函数的某些值,纵坐标表示执行时间(以秒计算)。本算法与 Apriori 算法执行时间与最小支持度的关系如图 1 所示。

从该图中,我们可以看出:在横坐标相同的条件下,基于等价类的关联规则算法比 Apriori 算法的执行时间更少;而且值得一提的是,在支持度阈值较大的情况下,基于等价类的关联规则算法与 Apriori 算法的执行时间相差不大;而在支持度阈值很小的情况下,前者与后者的执行时间相差很大,前者效率更高。

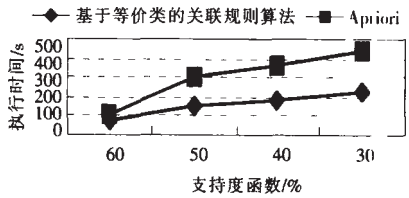

图 1 算法执行时间与最小支持度的关系

3 结论

我们在现有关联规则挖掘算法的基础上,基于等价关系和等价类以及文献[5]提供的算法,使得在生成侯选频繁项集和频繁项集时,算法执行时间和对原始交易数据库的扫描次数都有所减少。尤其当 k≥2 时,频繁 k-项集的数量明显减少,执行效率更高。(收稿日期:2005 年 6 月)

参考文献

1.Jiawei Han, Micheline Kamber.Data Mining: Conceps and Techniques[M]. Morgan Kaufmann Publishers, 2001-05

2.Agrawal R, Mannila H, Srikant R et al.Fast discovery of association rules: Advances in knowledge discobery and data mining [M].California: MIT Press, 1996: 307~328

3.Agrawal R, Srikant R.Fast algorithms for mining association rules in large databases[C].In: Proceedingsof the 20th International Conference on Very Large Databases, New York: Institute of Electrical and Electronics Engineers, 1994

4.王翔,袁兆山.基于等价类和最大完全图集聚类的关联规则发现算法[J].小型微型计算机系统, 2000; 21(6)

5.施润身,赵青.改进的关联规则采掘算法及其实现[J].同济大学学报, 2002; 30(2)

(上接 174 页)

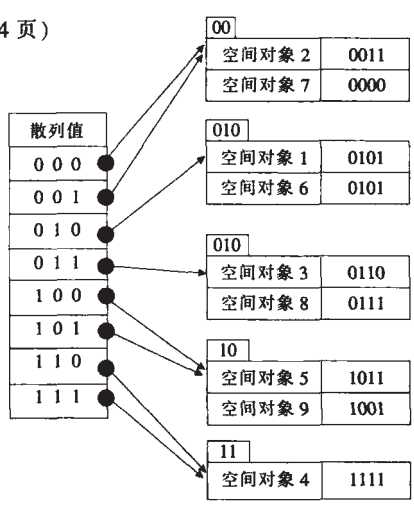

图 7 最终的散列结构

机运算的空间索引算法。该算法可用于构造二维空间数据的索引领域,实践表明该算法高效、简单,并具有很强的实际应用及推广价值。(收稿日期:2005 年 9 月)

参考文献

1.A Guttman.R-Trees a dynamic index structure for special search [C].In: Proc ACM S IGMOD, 1984: 47~57

2.Abraham Silberschatz 等著,杨东青等译.数据库系统概念[M].机械工业出版社, 2002

3.顾军,吴长彬.常用空间索引技术的分析[J].微型电脑应用, 2001; (12)

4.董鹏等.一种基于改进四叉树的 GIS 空间选择查询算法[J].计算机工程与应用, 2003; 39(13): 58~61

5.陈述彭等.地理信息系统导论[M].科学出版社, 1999

表 3 动态散列技术相对于四叉树的优点

比较项目	动态散列法	四叉树
空间对象的定位	计算其 X、Y 位置值,再通过位合并影射出二进制散列值	从地图全局窗口开始,通过空间关系判断,往下搜索,直至定位到叶结点
索引构造过程	采用"自下而上",散列出空间网格的散列值,再通过二进制的位运算,来构造索引	采用"自上而下",通过递归和空间对象的位置比较来构造四叉树
计算机运算效率	高,比较运算过程是二进制数的位运算,特别适合计算机语言实现	较差,比较运算过程是空间对象的位置运算,而这一过程要对空间对象的所有浮点型坐标进行比较才能得到结论,处理效率比动态散列法差的多

表 4 动态散列和四叉树的构造时间对比表

方 法	空间对象数目	空间对象类型	桶 长	运行时间/s
动态散列法	3 140	多边形	20	6.782
动态散列法	3 140	多边形	30	6.361
四叉树	3 140	多边形	20	90.124
四叉树	3 140	多边形	30	89.991