

P2PGIS 空间矢量数据动态 管理机制研究*

吴家皋¹, 田宇坤¹, 张曦¹, 邹志强¹, 胡斌²

(1. 南京邮电大学 计算机学院 计算机技术研究所, 南京 210003; 2. 南京师范大学 虚拟地理环境教育部重点实验室, 南京 210046)

摘要: P2PGIS 有效地解决了集中式 GIS 中海量数据的存储问题以及数据索引过程中产生的瓶颈效应。针对当前 P2PGIS 中由于节点不稳定, 容易造成数据丢失和索引失效的问题进行研究, 提出了 P2PGIS 空间矢量数据动态管理机制。该机制可以在分布式网络中对空间矢量数据进行多重备份, 并利用网络中存在的备份进行数据恢复、索引更新等操作, 有效地保证了数据的完整性及索引的正确性。同时, 通过优化路由索引策略进一步提高了索引效率。理论分析及实验结果表明, 采用空间矢量数据动态管理机制有效地提高了范围查询成功率, 缩短了数据索引时延。

关键词: 对等网络; 矢量数据; 路由策略; 数据管理; 索引网络

中图分类号: TP393

文献标志码: A

文章编号: 1001-3695(2012)10-3900-04

doi:10.3969/j.issn.1001-3695.2012.10.079

Study on dynamic management mechanism on space vector data in P2PGIS

WU Jia-gao¹, TIAN Yu-kun¹, ZHANG Xi¹, ZOU Zhi-qiang¹, HU Bin²

(1. Institute of Computer Technology, College of Computer, Nanjing University of Posts & Telecommunications, Nanjing 210003, China; 2. Key Laboratory of Virtual Geographic Environment, Ministry of Education, Nanjing Normal University, Nanjing 210046, China)

Abstract: P2PGIS has the potential to solve the mass data storage problem and can alleviate the bottleneck effect of data index in centralized GIS. Currently, the nodes in the network are not stable enough and likely to cause data missing and index disordering. This paper proposed a dynamic management mechanism on space vector data in P2PGIS. This mechanism could accomplish the vector data's multiple backup, recovered data or updated index with the backup data, so that ensured the data's integrity effectively. Furthermore, optimizing the routing policy could improve the index efficiency. Theoretical analysis and experimental results show that the dynamic management mechanism can improve query success rate and shorten the query latency.

Key words: P2P networks; vector data; routing policy; data management; indexing networks

0 引言

P2PGIS 是对等计算(P2P)技术与地理信息系统(GIS)结合的产物。该系统将地理信息数据分散存储在对等网络的节点上, 并利用 P2P 网络的路由机制完成相关数据的索引操作。P2PGIS 既能很好地解决空间矢量数据数据量大、计算复杂等问题, 又降低了对主机性能、网络带宽、投入成本的要求, 同时还提高了网络的灵活性和扩展性。

空间数据的组织和索引是 P2PGIS 首要解决的问题。文献[1]中提出了一种面向空间矢量数据的混合结构 P2P 空间索引网络(hybrid P2P spatial indexing network, HPSIN)。该网络将结构化 P2P 网络协议 Chord^[2]与空间矢量数据管理结构 quad-

tree^[3]相结合, 能同时兼顾系统负载的均衡和数据的高效索引。但是, P2P 网络中节点的不稳定性(如节点失效、退出以及新节点的加入等)会造成数据的丢失或索引的错误。

如何保证 P2P 网络拓扑结构的改变不影响节点中数据的完整性与索引的正确性已经成为 P2P 研究的热点。文献[4]提出将文件的 K 重备份保存在 Chord 网络中 K 个连续的节点上, 并利用备份数据分担原节点查询任务的策略; 文献[5]提出在 P2P 网络中添加中央服务器, 当节点数据丢失时通过访问中央服务器进行数据恢复; 文献[6]提出了对数据进行分片并分别保存在网络中不同节点上的备份机制。但是, 上述备份机制都是针对普通数据文件, 而针对空间矢量数据在 P2PGIS 中的数据备份与恢复策略的研究还鲜有报道。

收稿日期: 2012-03-12; **修回日期:** 2012-04-19 **基金项目:** 国家“863”计划资助项目(2009AA12Z219); 国家自然科学基金资助项目(40801149); 东南大学计算机网络和信息集成教育部重点实验室开放研究基金资助项目(K93-9-2010-06)

作者简介: 吴家皋(1969-), 男, 江苏苏州人, 副教授, 博士, CCF 会员, 主要研究方向为计算机网络协议及算法、P2P 网络、GIS(jgwu@njupt.edu.cn); 田宇坤(1987-), 男, 河北石家庄人, 硕士研究生, 主要研究方向为 P2P 网络及其应用、GIS; 张曦(1988-), 女, 江苏淮安人, 硕士研究生, 主要研究方向为 P2P 网络及其应用、GIS; 邹志强(1967-), 男, 江苏南京人, 副教授, 博士, 主要研究方向为分布式空间信息处理、P2P 网络; 胡斌(1975-), 男, 湖南衡阳人, 副教授, 博士, 主要研究方向为 P2PGIS、3D 建模。

为此,本文提出了一种新的 P2PGIS 空间矢量数据动态管理机制。利用本机制对空间矢量数据进行多重数据备份,保证了数据完整性。同时,利用备份数据为网络索引提供服务,通过改进路由协议优化了索引机制。

1 P2P 矢量数据混合结构索引网络

图 1 所示为本文所提出的 HPSIN 的结构,主要由上层的 Chord 网络和下层的 quadtree 结构组成。

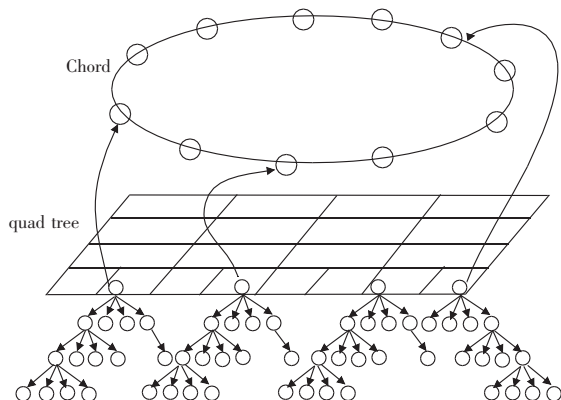


图 1 P2P 矢量数据混合结构索引网络

在 HPSIN 中,使用控制点将 quadtree 各层之间联系起来。所谓的控制点是指数据块最小外包矩形的中心点,因为 quadtree 上每个节点的地理范围不同,故可用控制点唯一标志 quadtree 上的每个节点。但将树型索引结构应用在查询中会导致根节点的查询访问量过大。因此,选取合适的最小初始层次($f_{\min}$),将 quadtree 划分为由若干子树构成的森林。每个子 quadtree 的根节点利用 hash 算法生成与 Chord 节点 ID (NID) 等长(假设利用 hash 算法生成的节点 ID 长度为 m ,则 Chord 网络所能容纳的节点数最多为 2^m 个)的数据 ID (DID)。

利用文献[7]中提出的数据分片方法,对空间矢量数据经过 WKT 转换器以及 JTS 转换器进行分片,每一个分片对应存储每个节点的空间矢量数据,以子 quadtree 根节点的 DID 作为 key 将 quadtree 上的所有节点数据分片发布/备份到 Chord 网络中的节点上,并利用键值对 $\langle \text{DID}, \text{FilePath} \rangle$ 记录数据块分片在节点中的存储位置。多个数据分片存储在一个节点上便构成了此节点的数据索引序列。

2 P2PGIS 空间矢量数据动态管理机制

本文提出的数据动态管理机制以空间矢量数据分片为对象,针对每一个分片进行动态数据管理,可以有效地防止数据备份、恢复以及索引更新过程中造成的系统不稳定问题。

2.1 数据发布/备份策略

为了避免节点失效导致数据丢失,本文提出的动态数据管理机制利用多重备份策略使 2^k 个完全相同的数据副本等间隔地分布在 Chord 网络的节点上。采用式(1)计算备份数据 ID (BDID) 并利用此 ID 将数据发送到 Chord 网络中对应的节点上。

$$\text{BDID}_i = (\text{DID} + i \times 2^{m-k}) \bmod 2^m \quad (1 \leq i \leq (2^k - 1)) \quad (1)$$

其中: DID 为发布数据的 ID, BDID_i 为原始数据的第 i ($i=1, 2, 3, \dots, 2^k-1$) 个备份数据的 DID。BDID _{i} 在 Chord 环上的位置与 DID 偏移量为 2^{m-i} , 当需要对分片数据进行插入、删除以及

更新等操作时,利用消息机制将操作内容发送给所有的备份节点,并对备份数据进行相应的操作。

2.2 节点失效时的数据恢复

采用数据动态管理机制对数据进行恢复时,虽然网络中存在多重备份,但在恢复时仅利用距离其最近的备份节点完成数据恢复操作。

图 2 是不失一般性的数据备份示意图。图中仅标志出了节点 x 与节点 x 距离为 $(x + 2^{m-k}) \bmod 2^m$ 所对应的节点(节点 x 所保存数据的最近备份节点)以及这两个节点周边节点的信息。其中,节点 x 的前驱节点为 y_1 , 后继节点为 y_2 。节点 x 上保存的数据块的 ID 为 DID。若节点 x 失效或退出网络,则会导致根据 hash 算法散列到本节点上的原始数据以及本节点保存的其他节点的备份数据丢失。

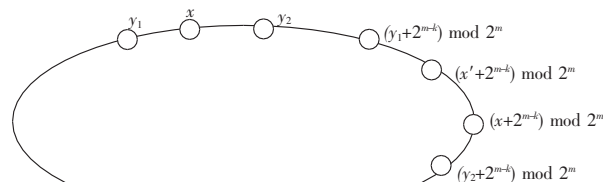


图 2 数据备份示意图

图中 $y_1 < x' < x < y_2$ 。由数据动态管理机制的数据发布/备份策略可知,节点 x 本身的数据以及备份的其他节点的数据都会存储在数据 ID (DID) 大于 $(y_1 + 2^{m-k}) \bmod 2^m$ 且小于 $(x + 2^{m-k}) \bmod 2^m$ 的顺序排列的节点上。因此,数据块的恢复过程需要操作从 $(y_1 + 2^{m-k}) \bmod 2^m$ 所对应的节点开始到 $(x + 2^{m-k}) \bmod 2^m$ 所对应的节点终止其间顺序排列的所有节点。为了标志数据恢复终止节点的位置,需要在每个节点中增加数据恢复终止标示符,在进行数据恢复前将终止节点的标示符设置为真。

数据恢复流程描述如图 3 所示(图中节点标志为该 DID 所对应的节点),其数据恢复过程为:

- 节点 y_1 检测到其后继节点 x 退出,发送消息给 $(x + 2^{m-k}) \bmod 2^m$ 所对应的节点将其数据恢复终止标志符设为真。
- 节点 y_1 向 $(y_1 + 2^{m-k}) \bmod 2^m$ 所对应的节点发送数据恢复请求。
- 节点接收到数据恢复请求,将本节点数据重新发布/备份到节点 y_1 新的后继节点,然后验证数据恢复终止标志符,若为假,则向其后继节点转发数据恢复请求;若为真,则结束操作。
- 循环执行 c) 直到结束操作,发送数据恢复完成消息给节点 y_1 。

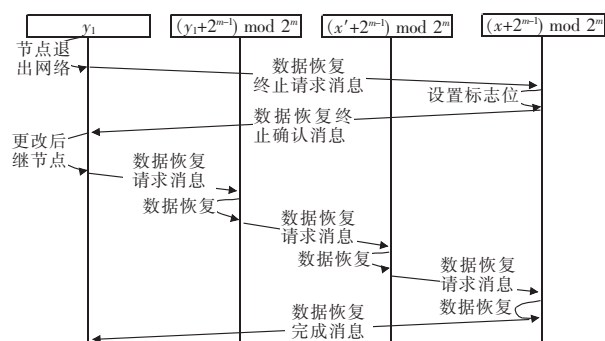


图 3 数据恢复流程

2.3 节点加入时的数据调整

若有新节点加入网络,则新加入节点的数据及其索引在其加入之前通过散列算法都保存在其后继节点上。验证其后继节点上的分片数据的 DID,若 $DID \bmod 2^{m-k}$ 小于新加入节点 $NID \bmod 2^{m-k}$,则此数据块应重新发布/备份到新加入节点上,将这部分数据发送给其前驱节点并在本节点上删除相关信息,其前驱节点保存相应数据并更新索引列表,完成数据更新。

2.4 空间数据的动态索引算法

进一步地,可通过备份数据所在的节点来协同完成索引服务。当节点 ID 为 NID 的节点查询数据 DID 时,可利用式(2)查找索引的目的节点:

$$(DID + 2^m - NID) \bmod 2^m > 2^{m-k}?$$

$$\text{search}(NID + (DID + 2^m - NID) \bmod 2^{m-k}); \text{search DID}; \quad (2)$$

由式(2)可知,DID 与 NID 差值大于 2^{m-k} 时,将数据查询请求发送给距离本节点最近的备份数据所在的节点,由备份节点响应查询请求完成数据索引操作。完整的空间矢量数据的索引过程如下:

算法 1 空间矢量数据范围查询的索引算法

```

query(Envelope env) { //env 为查询范围
//获取要查询的 quadtree 数据块的  $f_{\min}$  层控制点信息
ControlPointList controlPoints = getControlPoints(env);
//并行发送 quadtree 数据请求,id 为本节点的 NID
for each cp  $\in$  controlPoints do in parallel{
    Data data;
    if(cp.getDid +  $2^m - id$ )  $\bmod 2^m < 2^{m-k}$  {
        //从数据的原节点获取数据
        data = getDataFromChord(cp.getDid, cp);
    } else {
        //从距离本节点最近的备份节点获取数据
        data = getDataFromChord(
            id + (cp.getDid +  $2^m - id$ )  $\bmod 2^{m-k}$ , cp);
    }
    Datalist = Datalist  $\cup$  {data};
}
//对所得到的数据进行矢量合并处理操作
Result result = Merge(Datalist);
return result;
}

```

3 数据动态管理机制的性能分析

在 GIS 中范围查询是最基本的空间查询方式。范围查询是指利用界定的查询范围与空间矢量数据进行相交操作并返回交集部分所对应的所有数据的一种查询操作。交集部分的数据块就是本次范围查询所要索引的内容。下面从范围查询的成功率和延迟两方面就本文所提的方案作简要理论分析。

假设一次范围查询需索引的数据块(即 quadtree 上的控制点)的个数为 S ,设网络中节点的失效概率都为 $p(0 < p < 1)$ 。则 $k > 0$ 时数据动态管理机制网络范围查询成功率为

$$P_B = (1 - p^{2^k})^S \quad (3)$$

当 $k = 0$ 时的动态数据管理机制网络,即无动态数据管理机制的原始网络,其范围查询的成功率为

$$P_N = (1 - p)^S \quad (4)$$

显然有 $P_N < P_B$,即数据动态管理机制能够提高范围查询的成功率且 k 值越大范围查询成功率越高。同时,随着查询范围的增加,范围查询成功率将整体呈降低趋势。

由式(2)可知,若网络节点数为 N ,则采用动态索引算法的平均路由跳数为 $\log_2(N/2^k)$,比原始 Chord 网络的平均路由跳数 $\log_2(N)$ 减少了 k 跳。故采用数据动态管理机制能够缩短范围查询的延迟,提高索引效率。

4 原型系统仿真与分析

利用 PeerSim^[8] 对系统进行仿真测试。选取 $f_{\min} = 5$,即将整个空间矢量数据划分为 1 024 棵 quadtree 数据块,且每一棵 quadtree 维护整个矢量数据空间 1/1 024 的查询范围的数据。一次范围查询所需要查询的 quadtree 数据块的个数与查询范围的大小成正比。由于数据块 DID 是随机产生的,故范围查询越多所需要索引的节点数越多。

4.1 提高范围查询的成功率

根据式(3)可知数据动态管理机制可以提高范围查询的成功率。从实际应用出发,为空间矢量数据的查询范围引入比例尺(scale)的概念,比例尺的大小决定了查询范围的大小,且有:

$$\text{queryRange} = \frac{\text{totalRange}}{4^{\text{scale}}} \quad (5)$$

图 4 为在含有 1 000 个节点的 Chord 网络中,单节点失效概率为 10% 时, $k = 0$ (无数据动态管理机制)、 $k = 1$ 、 $k = 2$ 的动态管理机制网络在不同查询范围下进行范围查询失败概率的比较。从图中可知查询范围越大查询成功率越低,同时,利用动态数据管理机制在不同的查询范围下都可以有效地提高范围查询的成功率,且 k 值越大范围查询成功率越高。

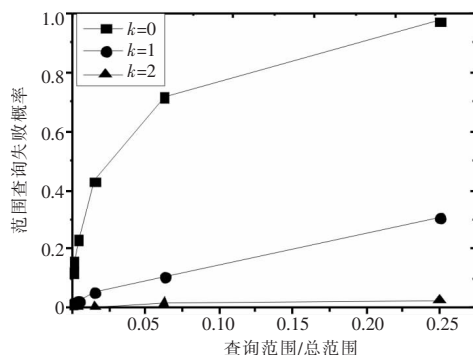


图 4 范围查询失败概率统计

图 5 为 $k = 1$ 时的数据动态管理机制网络中节点失效概率分别在 5%、10%、15% 的情况下进行不同查询范围的范围查询时的失败概率。由图可知,范围查询的成功率与查询范围、节点失效率呈正比例递增,但采用动态管理机制可以使范围查询的失败率呈指数形式递减。

4.2 减少数据查询的路由跳数

由式(2)的数据索引策略可知对于请求数据的节点与索引节点偏移量大于 2^{m-k} 的数据请求,向距离本节点最近的数据的备份节点响应数据查询请求。这样可以有效地减少数据索引过程中的路由跳数。

图 6 为在 1/4 查询范围进行 1 000 次范围查询时, $k = 0$

(无数据动态管理机制)、 $k=1$ 、 $k=2$ 时范围的平均路由跳数,由图可知动态管理机制可以有效地减少路由跳数,且 k 值越大,平均路由跳数越少。

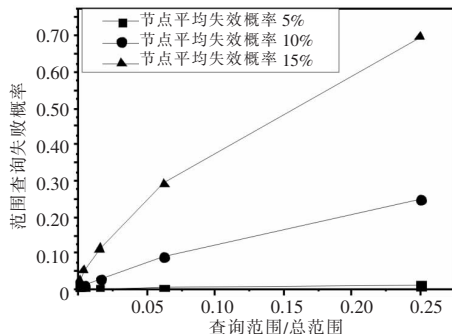


图 5 $k=1$ 时不同节点失效概率下范围查询失败概率统计

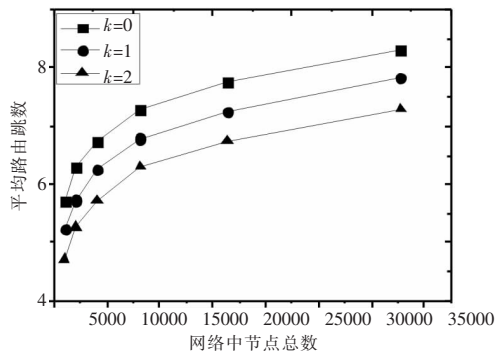


图 6 1/4 范围查询时的平均路由跳数

5 结束语

基于 P2P 网络巨大的优势和广阔的发展前景,本文在混合结构 P2P 空间索引网络的基础上提出了一种数据动态管理机制,对已有的研究作了一些改进。系统的原型已经完成,实现了对数据的备份以及索引的优化。但是,在目前的研究中只

对数据备份的优势进行了分析,数据备份同样也伴随着巨大的网络和节点开销。考虑到数据的可靠性要求以及分布式网络的规模,笔者将在今后的研究中对采用 2^k 重备份数据管理机制中 k 的最优值进行论证求解。

参考文献:

- [1] WU Jia-gao, JIANG Nan, ZOU Zhi-qiang, *et al.* HPSIN: a new hybrid P2P spatial indexing network [J]. *The Journal of China Universities of Posts and Telecommunications*, 2010, 17(3): 66-72.
- [2] STOICA I, MORRIS R, KAASHOEK M F, *et al.* Chord: a scalable peer-to-peer lookup service for Internet applications [C]//Proc of ACM SIGCOMM. New York: ACM Press, 2001: 149-160.
- [3] TANIN E, HARWOOD A, SAMET H. A distributed quadtree index for peer-to-peer settings [C]//Proc of the 21st International Conference on Data Engineering. Washington DC: IEEE Computer Society, 2005: 254-255.
- [4] ANGLANO C, FERRINO A. Using Chord for meta-data management in the N3FS distributed file system [C]//Proc of International Workshop on Hot Topics in Peer-to-Peer Systems. Washington DC: IEEE Computer Society, 2004: 96-101.
- [5] SIT E, CATES J, COX R. A DHT-based backup system [EB/OL]. http://doc.cat_v.org/plan_9/misc/venti-dhash/sit.pdf.
- [6] LIU Tian-hua, BAO Ming-yu, CHANG Gui-ran, *et al.* The improved research of Chord based on file-partition replication strategy [C]//Proc of the 9th International Conference on Hybrid Intelligent Systems. Washington DC: IEEE Computer Society, 2009: 411-414.
- [7] 吴家皋, 黄琳, 邹志强, 等. 基于分布式拓扑的 P2P 矢量地理数据表示模型的研究 [J]. *计算机应用研究*, 2010, 27(6): 2170-2172, 2180.
- [8] LEBERTO M, MARK J. PeerSim: a scalable P2P simulator [C]//Proc of the 9th International Conference on Peer-to-Peer Computing. 2009: 99-101.