基于对等计算的分布式时空查询处理系统的设计及应用研究 *

杨晓亮, 张 翀, 汤大权, 肖卫东, 唐九阳
(国防科学技术大学 信息系统工程重点实验室, 长沙 410073)

摘 要: 采用对等计算技术解决分布式时空查询处理问题,分析并设计了基于对等计算的时空查询处理系统。参与计算的节点通过合理有效的拓扑结构进行组网,整体上对外提供时空查询服务,查询负载被合理地分配到相关节点中,大大增强了系统的鲁棒性和容错性,提高了系统性能和抗毁能力。最后通过应用实例说明系统具有很大的应用价值。

关键词: 时空查询处理; 对等计算; 设计; 应用实例研究
中图分类号: TP311 **文献标志码**: A **文章编号**: 1001-3695(2011)04-1342-06
doi:10.3969/j.issn.1001-3695.2011.04.041

Distributed spatio-temporal query processing system based on peer-to-peer computing: design and case study

YANG Xiao-liang, ZHANG Chong, TANG Da-quan, XIAO Wei-dong, TANG Jiu-yang
(*Science & Technology on Information Systems Engineering Laboratory*, *National University of Defense Technology*, *Changsha* 410073, *China*)

Abstract: This paper proposed a novel solution which leveraged peer-to-peer computing to accomplish distributed spatio-temporal query processing, and designed a spatio-temporal query processing system based on P2P. Nodes were organized into effective topology and they cooperatively provided spatio-temporal query ability. So query load was distributed among the nodes properly, which greatly reinforced system robustness and fault tolerance and highly enhanced performance and invulnerability. Finally, discussed case studies to illustrate the applications of the system.

Key words: spatio-temporal query processing; peer-to-peer computing; design; case study

0 引言

随着无线通信技术、定位技术、嵌入式设备以及地理信息系统(geographic information system,GIS)的快速发展和普及,获取并存储时空信息成为可能。基于时空信息可以建立位置服务,这可以帮助用户快速进行时空查询并获取对象,完成与时空相关的活动,其应用范围渗透至导航系统、气象监测系统、交通控制系统、土地信息系统、移动计算、媒体系统、人口统计及病例跟踪等各个领域[1]。特别地,在数字化战场中,获取并合理利用时空数据进行战场态势感知可以遂行有效的警戒行动、战场环境保障以及支持武器单元进行精确打击等任务,对提高战场透明度、提高联合作战能力,实施高效的指挥决策具有重要意义。

时空数据库[2]是近几年数据库领域内一项新兴技术,它以有效管理时空信息为目的,存储并索引大量的时空数据,对外提供快速的时空查询服务。时空查询处理技术是时空数据库领域内的一项重要研究内容,它通过设计合理的时空索引和查询算法,并结合有效的预处理,对复杂的时空查询分解、分析并对记录进行搜索,最后快速地将结果返回给用户。时空查询处理技术的强弱反映了时空数据库性能的优劣,因此,国内外大量的研究学者给予了极大关注。

然而,随着采集手段的不断发展,时空数据成海量式堆积,这直接推动了时空查询处理技术从集中式走向分布式。集中式的时空查询处理会带来一系列弊端:

a)数据与查询的负载都集中在一个中心,数据量的增大和查询用户的增多会严重地影响系统性能;

b)若中心失效,则无法提供时空查询服务;

c)组织模式不够灵活,无法支持大规模广域应用。

本文提出采用对等计算(peer-to-peer computing,P2P)技术架构实现分布式的时空查询处理,P2P 模式比传统的分布式模式更加适合于建立分布式时空查询处理系统,原因如下:a)从系统性能的角度,服务器/客户端(C/S 模式)和层次式等模式具有瓶颈、"单点失效"等缺点,无法适应动态变化的环境,而 P2P 模式本身对等、无中心的特点使得整个系统能力随着节点的增加而增强,大大地提升了系统可靠性;b)从系统互操作性的角度,C/S 模式不适合大规模的协作需求,而 P2P 模式将极大地增强节点之间的协作能力,提高共享力度;c)从系统灵活性的角度,P2P 模式的自治性和动态性将更加大幅度提高整个系统的自适应性。

收稿日期: 2010-09-19; **修回日期**: 2010-10-21 **基金项目**: 国家自然科学基金资助项目(60172012)

作者简介:杨晓亮(1986-),男,河南平顶山人,硕士研究生,主要研究方向为分布式时空索引(kimyxl@163.com);张翀(1982-),男,博士研究生,主要研究方向为对等计算、时空索引;汤大权(1971-),男,教授,硕导,博士,主要研究方向为信息资源管理;肖卫东(1968-),男,教授,博导,博士,主要研究方向为信息资源管理;唐九阳(1978-),男,副教授,博士,主要研究方向为对等计算、信息组织与检索.

基于 P2P 的时空查询处理系统是在分布式时空数据库这个背景下思考提出的,本文给出了系统的设计思路、整体和详细架构以及应用研究,为实现基于 P2P 的海量时空数据管理做出基础性工作。

1　相关工作

1.1　集中式时空查询处理

从时间轴上划分,时空查询处理分为面向历史的查询处理和面向将来预测的查询处理。

HR 树[3](historical R-trees)是一种面向历史信息的时空索引。它基于重叠技术将单一版本的数据结构转换为部分持久的结构。这种结构针对每一个时间戳维护一个 R 树结构,但连续的两个树结构之间的公共分支则仅存储一次以节约空间。HR 树对时间戳查询效率较高。但由于 HR 树本身的版本冗余的特点,造成了存储利用率不高,时间区间查询效率较差。

TPR* 树[4](time parameterized R-tree)是对 TPR 树的改进,它建立在 R* 树基础上,使用时参限定矩形的概念,能持续地随着其包围的移动对象移动或变化。TPR* 树索引节点记录存储移动对象 MBR 和 MBR 的速度矢量。TPR* 树基于 R* 树并利用时参的度量准则进行删除、插入及更新等动态操作。TPR* 树对静态的区域查询进行优化,对上溢节点分裂算法,在必要的时候才对节点分裂并重插。

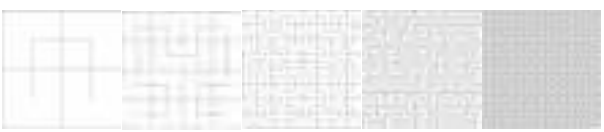
1.2　Chord

P2P 是一种新的分布式计算模式,它强调参与计算的各个节点独立、自治,各个节点自组织共同完成分布式计算任务,每个节点既对外提供计算服务,同时也享受其他节点提供的服务。

Chord[5]是 2001 年由麻省理工大学提出的一种结构化 P2P。Chord 运用一致性哈希函数将每个参与计算的节点和资源命名为一个长为 m 位的二进制串,即是一个 $0 \sim 2^m$ 的任意整数。为确保高效的路由与搜索性能,Chord 中每个节点除了维护前驱与后继邻节点之外,还维护一个名为 finger table 的路由表。Chord 利用一致性哈希技术破坏了原始空间的顺序,不能支持范围查询和时空查询。

1.3　Hilbert 曲线

Hilbert 曲线[6]是一种空间填充曲线,该曲线采用分形技术,通过递归构造将 d 维空间中每个格子的中心点进行连接,且进出每个格子仅一次,从而将 d 维的空间目标映射为 1 维目标。空间填充曲线具有较好的局部保持(locality-preserved)特性,即 d 维空间中对象被映射到 1 维空间后,在 1 维空间中邻近的对象,在原 d 维空间中也是邻近的,但反之则不成立。图 1 展示了 1 ~ 5 阶的 Hilbert 曲线。



(a) 1阶　(b) 2阶　(c) 3阶　(d) 4阶　(e) 5阶

图1　Hilbert曲线示例

整个空间采用 Hilbert 曲线划分为相等的格子,即 cell,经 Hilbert 曲线的划分,每个 cell 都会获得唯一的标志 ID,再使用 Chord 管理 1 维空间的机制,这样就完成了从多维空间向 1 维空间的转换,即整个空间中的每个 cell 在 Chord 空间中都有唯一的位置。根据 Chord 的机制,每个参与计算的节点维护从前驱节点到自己的一段区间,即维护整个空间中相应的 cell。

2　系统整体设计

系统的结构如图 2 所示,主要包含时空信息采集子系统、时空对象注册中心、将来预测查询处理子系统、历史查询处理子系统、用户查询子系统。系统的核心组成是将来预测查询处理子系统和历史查询处理子系统。

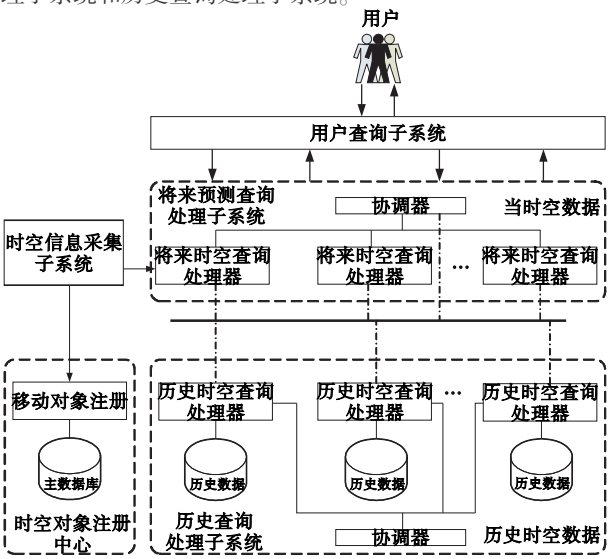


图2　系统整体设计图

时空信息采集子系统实时采集移动对象的相关信息,并将采集到的信息传送给将来预测查询处理子系统和时空对象注册中心。时空对象注册中心将采集到的移动对象信息进行注册,并将信息存储在数据库中。

将来预测查询处理子系统由一系列分布式计算节点组成,每个节点称为将来时空查询处理器。协调器动态调整每个节点处理对象的数目,使每个节点处理的对象数目达到动态平衡。每个节点存储时空对象当前的时空信息,由于对象的数据更新较快,每个节点都要有很高的更新速度。根据对象的时空信息,通过将来预测查询处理子系统可以预测未来某个时刻的对象信息。对于过期数据,将来预测查询处理子系统把这些数据转移至历史查询处理子系统。这样可以保证将来预测查询子系统所处理的时空数据是最新的,并且数据量较小,可以直接存储在缓存中,这样可以大大提高查询效率。将来时空查询处理器组成 Chord 网络,用于存储和索引当前时空信息,并回答将来预测查询。

历史查询处理子系统也是由一系列分布式计算节点组成,每个节点称做历史时空查询处理器,每个节点对应一个数据库,用于存储移动对象的历史时空信息。协调器把将来预测查询处理子系统转移过来的数据动态分配到每个节点中,使每个节点达到负载均衡。通过历史查询子系统,用户可以查询对象的历史信息。同样,历史时空查询处理器也组成 Chord 网络,用于存储和索引过去的时空信息,并回答历史查询。

用户查询子系统是提供给用户的查询接口,通过图像化的展示提供给用户友好的查询界面,用户通过查询子系统可以方便地查询所关心对象的历史位置信息和将来可能的位置信息,并通过图像化界面向用户展示查询结果。

3　系统详细设计

以下分各个子系统进行详细介绍，其中存储与索引机制、将来预测查询处理和历史查询处理是系统的核心，进行重点描述。

3.1　时空信息采集子系统

时空信息采集子系统由各种传感器组成，传感器定期采集数据并回传进行存储。传感器因应用的不同而不同。例如在采集对象位置信息时，传感器可能是定位卫星、成像卫星、无人机、带有定位功能的无线传感器网络等；在采集自然地理对象的变化信息时，如在地籍管理、大陆架迁移等应用中，传感器可能是测绘卫星、人工野外测绘仪等设备；在感知环境信息时，传感器可能是温度传感器、湿度传感器、热成像传感器、声音传感器等。

经由时空信息采集子系统发送回的记录统一规范为七元组(oid, v_x, v_y, p_x, p_y, shp, t)。其中，oid 为时空对象的标志，v_x 和 v_y 分别为该对象 t 时刻 x 方向和 y 方向的速度，p_x 和 p_y 分别为该对象 t 时刻的 x 坐标和 y 坐标，shp 为该对象 t 时刻的形状信息，可以由一系列坐标点表示。

3.2　时空对象注册中心

时空对象注册中心从全局角度出发，存储除时空属性以外的时空对象的相关属性信息。每个时空对象都自动地或人为地到注册中心进行注册，在查询处理时，查询结果可以参考注册中心的数据获取详细信息。注册中心存储的对象属性因不同应用而不同，如在移动对象位置管理中，可以存储对象的标志属性(ID)、对象名称、对象性质等；在地籍管理等应用中，可以存储对象的归属信息等。

3.3　存储与索引机制

针对采集子系统所获取的时空数据，如何进行合理存储是进行时空查询处理的关键所在。利用 Hilbert 曲线由高维向一维映射的机制，可以将时空信息都映射到一维空间中，而 Chord 机制恰好可以对一维数据进行有效组织与检索，这就是存储的基本机制。具体过程如下：

a)设整个待考察的空间为边长等于 E 的正方形(考虑二维空间，三维及以上可以类推)，并采用 D 阶 Hilbert 曲线，整个空间被划分为 2^{2D} 个大小相同的 cell。通过构造 Hilbert 曲线，存在从整个空间所有点的集合 P 到 Hilbert 值集合 V 的映射 H: $P \rightarrow V$，其中 $P = \{(x,y) \mid 0 \leqslant x, y \leqslant E\}$，$V = \{v \mid 0 \leqslant v \leqslant 2^{2D} - 1, v \in Z\}$。

b)使用 Chord 作为通信拓扑结构，其键值空间为 $[0, 2^m - 1]$，设 $m = 2D$，即键值空间与 Hilbert 值集合 V 一致。

c)将来预测查询处理子系统中的存储与索引方式。将来预测查询处理子系统采用基于内存的存储方式。因为经由采集子系统不断回传数据，将来预测查询处理子系统需要不断刷新，维护最新的时空数据，并定期将过期数据转移至历史查询处理子系统。与历史查询处理相比，将来预测查询处理子系统更关注时空对象的位置信息，而对形状可以不加考虑，即仅存储(oid, v_x, v_y, p_x, p_y, t)六元组。基于内存的存储方式、定期转移历史数据的机制以及精简存储内容可以降低存储开销，大大缩短用户的查询响应时间。当采集子系统将一条记录传输至某个节点 Q_f 时，Q_f 根据映射 H 将(p_x, p_y)映射为 Hilbert 值 v，Q_f 根据 Chord 协议查找键值 v 的后继节点 S_f，将该条记录发送至 S_f，由 S_f 负责存储该记录。每个参与组网的节点在本地建立基于内存的 TPR* 树，索引时空信息记录。

d)历史查询处理子系统中的存储与索引方式。历史查询处理子系统采用基于磁盘的存储方式。因为对于历史信息来说，空间粒度与时间范围是主要考虑的因素，空间粒度越细，时间覆盖范围越广，对用户分析历史情况越有利。因此历史数据往往规模庞大，需要依靠磁盘进行存储。当将来预测查询处理子系统将过期数据记录转移到对应的节点 Q_h 时，Q_h 首先在本地存储(oid, shp, t)三元组，并根据 shp 数据构造时空对象的最小限定矩形 MBR(X_l, X_u, Y_l, Y_u)，其中(X_l, Y_l)表示矩形左下角坐标，(X_u, Y_u)表示矩形右上角坐标。MBR 可能跨多个 cell，因此根据 Hilbert 映射规则，MBR 可能被映射为多个区间。设每个区间形式为[H_{enter}, H_{exit}]，Q_h 依据 Chord 协议查询键值 Hexit 的后继节点 S_h，并发送七元组(oid, X_l, X_u, Y_l, Y_u, t, IP_{Qh})至 S_h，由 S_h 对该记录进行索引，其中 IP_{Qh} 表示存储时空对象详细信息的节点 Q_h 的 IP 地址。每个存储(oid, shp, t)记录的节点在本地建立 HR 树时空索引。

例 1　针对历史查询处理子系统中的存储与索引方式举例进行说明(将来预测查询子系统相应例子与此类似，在此不再赘述)。图 3(a)所示为时空对象 $O_1 \sim O_5$ 分别在 $t = 10$，$t = 20$ 和 $t = 40$ 时的 MBR。采用 3 阶 Hilbert 曲线，整个空间被划分为 64 个 cell 且根据 Hilbert 曲线进行编号。图 3(b)所示为时空对象在各个节点中的存储与索引情况，如时空对象 O_5 在 $t = 20$ 时经 Hilbert 曲线映射后的区间分别为[43, 45]，[39, 40]和 34。图 3(a)中分别采用不同的阴影图案区别不同的区间，根据每个区间的 H_{exit} 值，$t = 20$ 时 O_5 的索引信息发布到了节点 N_{38}、N_{42} 和 N_{48} 中。图 3(b)中示例了节点 N_{21} 的 HR-树索引和节点 N_8 的 finger table。

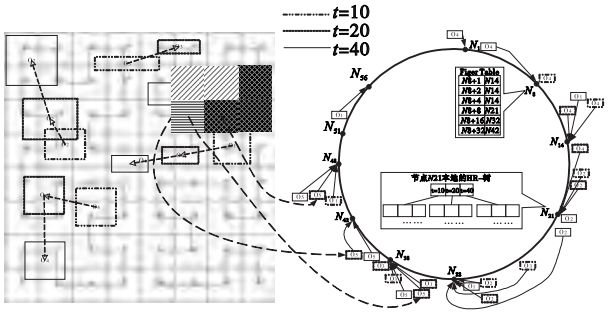


图3　索引结构示例

3.4　将来预测查询处理

将来预测查询处理子系统能够处理窗口查询、范围查询、KNN 查询以及连续查询等。以下主要描述窗口查询的处理过程。

给定一个面向将来的窗口查询(qx_l, qx_u, qy_l, qy_u, t_q)，其中(qx_l, qy_l)为查询窗口的左下角坐标，(qx_u, qy_u)为右上角坐标，t_q 为未来的查询时刻，返回在未来 t_q 时刻矩形(qx_l, qx_u, qy_l, qy_u)内的全部对象。将来预测查询处理中最核心的机制是根据目前的速度扩大查询窗口，将未来所有可能进入该窗口的对象全部囊括。设当前时刻为 t_n，节点 Q 发起窗口查询(qx_l, qx_u, qy_l, qy_u, t_q)，具体处理过程如下：

a)节点 Q 根据全局的最大速度 $Vel_{\max}$，将查询窗口的每个边以 $Vel_{\max}$ 为速度向外扩大，即对于查询窗口 $q = (qx_l, qx_u, qy_l,$

qy_u),扩展为 $q' = (qx_l' = qx_l - |Vel_{\max}|\Delta t,\ qx_u' = qx_u + |Vel_{\max}|\Delta t,\ qy_l' = qy_l - |Vel_{\max}|\Delta t,\ qy_u' = qy_u + |Vel_{\max}|\Delta t)$。其中 $\Delta t = t_q - t_n$;q' 是 q 能够扩大的上限。

b)节点 Q 计算四个矩形,分别为 rect_west = $(qx_l', qx_l, qy_l', qy_u')$,rect_north = $(qx_l', qx_u', qy_u, qy_u')$,rect_east = $(qx_u, qx_u', qy_l', qy_u')$,rect_south = $(qx_l', qx_u', qy_l', qy_l)$。分别计算与每个矩形相交的 cell 集合,分别记为 cell_west_list、cell_north_list、cell_east_list 和 cell_south_list。在 cell_west_list 中选取 x 正方向最大速度 $Velx_{\max}$,在 cell_north_list 中选取 y 负方向最大速度 $Vely_{\min}$,在 cell_east_list 中选取 x 负方向最大速度 $Velx_{\min}$,在 cell_south_list 中选取 y 正方向最大速度 $Vely_{\max}$。再根据此四个速度值,将 q 对应的四个边扩展为最终的查询窗口 Q_{enlgm}。

c)节点 Q 计算与 Q_{enlgm} 相交的所有 cell,并映射为一系列一维查询区间$[H_{\text{enter}}, H_{\text{exit}}]$,以 H_{exit} 为键值投递至 Chord 中进行查询。

d)收到查询请求的目的节点 S 执行基于 TPR* 树的本地搜索,并进行判断:若节点 S 的前驱节点 R 所在键值大于或等于收到的查询区间的 H_{enter} 值,将此查询请求转发给节点 R,R 继续执行步骤 d)。

例 2　如图 4 所示,查询窗口为 Q,根据全局最大速度,将 Q 扩大为 Q',再计算四个矩形{rect_west,rect_north,rect_east,rect_south}。依据四个矩形内的正方向和负方向的最大速度,将 Q 最终扩展为 Q_{enlgm}。计算与 Q_{enlgm} 相交的 cell 总共为 15 个,转换区间分别为 6,[9,10],28,[31,35],[52,57],将区间投递至 Chord 进行一维区间查询。以区间[52,57]为例进行说明,以该区间的 H_{exit} 值 57 为键值查询后继节点,结果为 N_1;然后 N_1 进行本地查询,并且 N_1 检查其前驱节点为 N_{56},所在键值大于该区间的 H_{enter} 值 52,符合查询条件;N_1 将查询转发给 N_{56},N_{56} 在本地进行查询,N_{56} 不必转发给其前驱节点 N_{51},对于查询区间[52,57]来说,查询停止。

关于如何获取全局最大速度,每个节点可以通过维护二维速度直方图来实现。由于所采用的通信结构为 Chord,可以利用 Chord 的心跳协议来捎带所存储时空对象的速度信息,这种模式称为 piggyback 模式。这样经过若干周期,每个节点对全局的速度分布情况大致有一定的了解,可以建立起针对每个 cell 的二维速度直方图,从而在查询时既可以获取全局最大速度,也可以获取某个 cell 中时空对象的最大、最小速度。

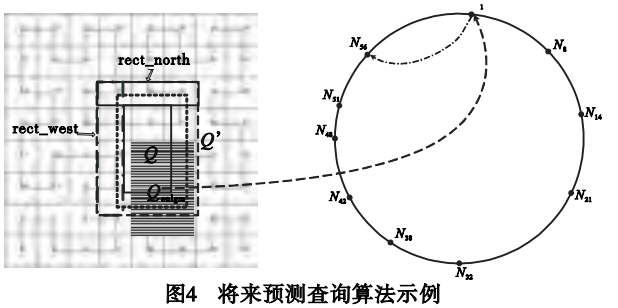


图4　将来预测查询算法示例

3.5　历史查询处理

历史查询处理子系统的设计思路与将来预测查询处理子系统相似,但在查询时不必扩大查询窗口。因为在将来预测查询处理子系统中,是对将来尚未发生的时空事件进行查询,因此需要将查询窗口依据将来的时间进行相应的扩大以包含未来可能进入查询窗口的对象;而对于历史查询处理来说,对象过去的信息都进行了存储,只需要查找到当时的时刻或时间区域进而实施空间查询即可,不必扩大查询窗口。历史查询处理子系统能够处理窗口查询、范围查询、KNN 查询以及轨迹查询。以下以历史窗口查询处理过程为主进行描述。

给定一个历史窗口查询$(qx_l, qx_u, qy_l, qy_u, t_q)$或$(qx_l, qx_u, qy_l, qy_u, t_s, t_e)$,其中,$(qx_l, qy_l)$为查询窗口的左下角坐标,$(qx_u, qy_u)$为右上角坐标,$t_q$ 为过去查询时刻,(t_s, t_e)为过去的查询时间区间,返回在 t_q 时刻或(t_s, t_e)时间段内矩形(qx_l, qx_u, qy_l, qy_u)内的全部对象。节点 Q 发起查询,具体处理过程如下:

a)节点 Q 根据 Hilbert 曲线的映射规则,将矩形(qx_l, qx_u, qy_l, qy_u)转换为一系列查询区间$[H_{\text{enter}}, H_{\text{exit}}]$。

b)针对每一个查询区间,节点 Q 根据 Chord 一维查询算法,以 H_{exit} 为键值投递至 Chord 中进行查询。

c)收到查询请求的目的节点 S 执行基于 HR 树的本地历史搜索,并进行判断:若节点 S 的前驱节点 R 所在键值大于或等于收到的查询区间的 H_{enter} 值,将此查询请求转发给节点 R,R 继续执行步骤 c)。

例 3　图 5 所示为时空查询算法的举例。与空间查询区域交叠的 cell 数量为 8,转换的查询区间分别为[33,34]、[45,46]、[50,52]和 55,不同的区间分别采用不同的阴影图案加以区别。根据这四个区间,查询条件分别会路由至节点 N_{38}、N_{48} 和 N_{56},节点 N_{56} 经比较再把查询条件转发给节点 N_{51},每个节点在本地根据 HR-树的时间点查询或时间段查询算法进行搜索,最后把结果返回给查询发起节点。

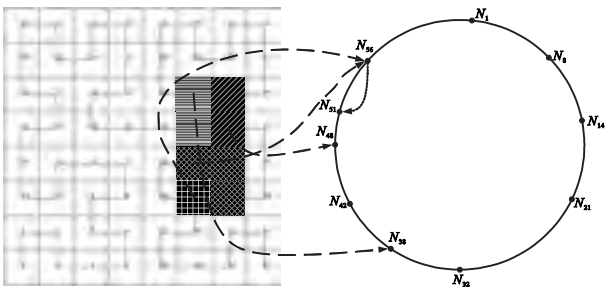


图5　历史查询算法示例

4　性能评价

为了验证本文所提方法的性能,使用意大利博洛尼亚大学的开源仿真平台 PeerSim[7] 进行仿真实验。所有实验都是在一个具有 Intel Core2 Quad 2.5 GHz 和 2 GB 主存的计算机上完成;整个待考察空间设为 1×1 的单位矩形,时间长度设为 1 的单位时间区间;由于实验中采用的矩形边长皆远小于待考察空间的边长,实验中默认采用 7 阶 Hilbert 曲线。

本文实现了 GSTD[8] 时空数据生成方法,该方法在进行时空索引的实验中被广泛采用。基于 GSTD 方法生成了实验数据集。基态数据为中心点呈 $\mu = 0.5, \sigma = 0.1$ 的高斯分布,边长呈(0,0.1)均匀分布的 N_d 个矩形数据;时间采样点数量为 100,且均匀分布;中心点的变化量(Δcenter)呈(−0.2,0.2)均匀分布,边长的变化量(Δextent)呈(−0.05,0.05)均匀分布。随着时间的推进,数据的整体变化为从待考察空间的中心向四周扩散。图 6 展示了 $N_d = 1\,000$ 时,时刻 t 分别为 0、0.25、0.5、0.75 和 1 时的所有数据。

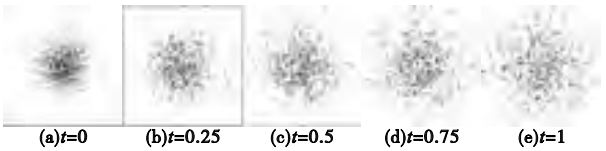


图6　数据集

本文进行了如下实验：

a)响应时间。将本文的系统与集中式的 HR 树和 TPR* 树作查询响应时间对比。设定查询的空间窗口的选择率为 0.05，节点数量为 1 000，不断增加数据规模 N_d，从 1 000 至 100 000，测试响应时间。实验结果如图 7 所示。

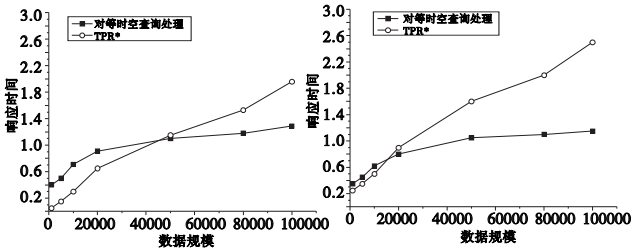


图7　响应时间对比

b)查询代价。设定查询的空间窗口的选择率 N_d 为 0.05，数据规模 N_d 为 5 000，增加节点数量从 250 至 10 000，测试查询引起的消息数量。实验结果如图 8 所示。

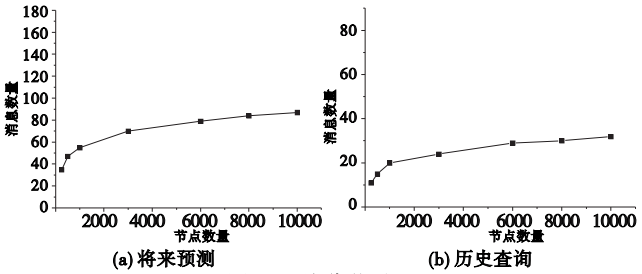


图8　查询代价对比

c)窗口查询。设定数据规模 N_d 为 5 000，节点数量为 1 000，变化查询的空间窗口的选择率从 0.001 至 0.07，测试查询引起的消息数量。实验结果如图 9 所示。

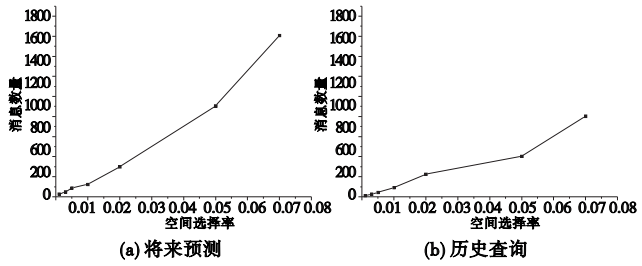


图9　消息数量对比

由图 7 可知，随着数据规模的不断增大，无论是将来预测查询还是历史查询，基于对等计算的时空查询处理技术的响应时间要低于对应的 TPR* 树和 HR 树，并且响应时间与数据规模的变化成对数关系，说明基于对等计算的时空查询处理技术具有较好的可扩展性。

由图 8 可知，从网络规模的角度度量，基于对等计算的时空查询处理技术具有良好的可扩展性，随着网络规模不断扩大，查询所引起的消息数量呈对数增长。

由图 9 可知，随着空间选择率的不断增大，将来预测查询所引起的消息数量要增长得更快，这是因为将来预测查询中需要扩大查询窗口，这会引起更多的消息进行查询。

5　应用研究

由于分布式时空查询处理系统具有较强的自治性、灵活性和抗毁性，在军事和民用领域有广泛的应用。

5.1　信息化联合作战战场监控

信息化条件下，网络化的信息系统将陆、海、空、天、电构成多维一体战场，夺取战场信息优势可以借助卫星、无人机、地面传感器等设备采集关于战场的实时信息，快速有效地实现信息的共享，获取战场信息资源成为作战决策中的重要支撑。如图 10 所示，战场中各参战军种、各级指挥机构及单兵之间都能实现信息的高度共享。

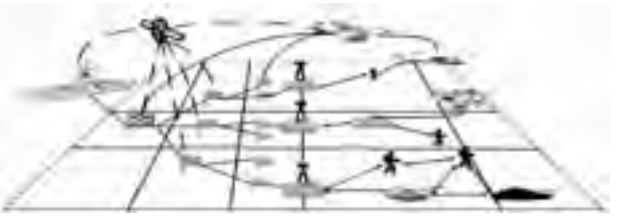


图10　信息化联合作战战场监控示意图

假设在某次联合作战中有三支参战力量，分别为 A：陆军、B：海军、C：空军。A 主要侦查地面战场的情报，监控地面战场态势的变化，其侦查手段和方式主要有侦查分队、无人机和地面传感器。B 主要侦查海面战场的情报，监控海面战场态势的变化，其侦查手段和方式主要有卫星、地面雷达。C 主要侦查空中战场的情报，监控空中战场态势的变化，其侦查手段和方式主要有预警机、地面雷达。三支参战力量所侦查的情报通过分布式时空查询处理系统进行管理。

例如，在 15:00 时刻 A 可以通过分布式时空查询处理系统查询此时距海岸线 40 海里范围内有多少敌方舰艇，并可以预测在 15:30 时刻有多少敌方舰艇。同时 A 也可以查询此时有多少敌方飞机向自己所在阵地飞来，多长时间之后可能对自己构成威胁。B 可以查询在敌方某区域内有多少阵地防空力量和机动防空力量，并可以预测敌方机动防空力量的动向。

5.2　传感器网中分布式时空查询

随着通信技术和传感器的迅速发展，世界范围的传感器网(world-wide sensor Web，WSW)的设想即将成为现实。通过建立统一的时空信息基础设施，分散的传感器网络(sensor network)将组成一个整体的时空传感器网(spatio-temporal sensor Web，SSW)。整个时空传感器网的基础架构如图 11 所示。

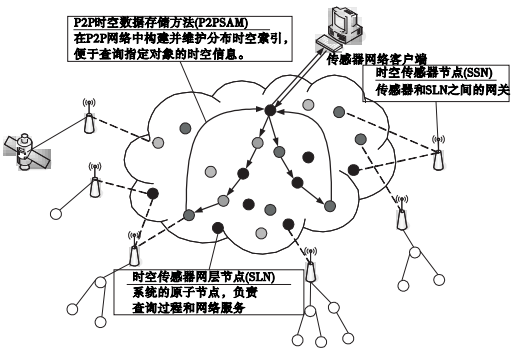


图11　传感器网络分布式查询系统结构图

系统结构主要由以下四部分组成：

a)时空传感器网层(spatio-temporal sensor Web layer,

SSL)。理论上讲,系统是由多个 SSW 层构成,每个 SSW 层针对不同的需求提供不同的服务,如 SSW 的气温层可以提供气温数据。每个层都是由相同类型的 SSW 节点组成。

b)时空传感器网层节点(spatio-temporal sensor Web layer node,SLN)。系统由多个 SLN 组成,每个 SLN 以 P2P 的方式进行网络连接,并提供时空信息查询。每个 SLN 存储着不同位置的时空信息,通过分布式索引结构对信息进行索引。当用户进行查询时,通过建立的索引结构进行信息查询,提高了查询效率。

c)时空传感器节点(spatio-temporal sensor node,SSN)。SSN 是传感器节点和 SLN 之间的网关,在 SSW 中扮演数据提供者的角色。每个 SSN 从传感器中收集原始数据,并发送给一个或者多个 SLN。

d)系统客户端。客户端可以给用户提供二维或者三维的地图,用户通过客户端可以查询自己所需的相关信息。

在时空传感器网中,传感器将采集到的时空信息存储于基站中,不同的基站经底层网络连通相互通信,通过引入基于 P2P 的时空查询处理机制,可以极大方便用户进行全局时空传感器网的时空查询。例如用户可以通过系统的客户端查询某块区域在某时刻的温度、湿度和光照情况。

5.3　物联网

物联网[9](Internet of things)是通过射频识别(RFID)、红外感应器、全球定位系统、激光扫描器等信息传感设备,按约定的协议,把任何物品与互联网连接起来,进行信息交换和通信,以实现智能化识别、定位、跟踪、监控和管理的一种网络,其结构如图 12 所示。

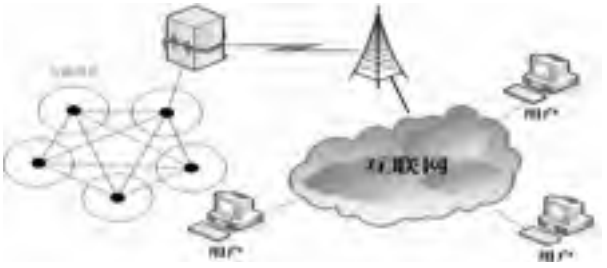


图12　物联网示意图

在物联网中,各传感器节点分布式部署在各地,采集的信息也分布式存储在相应的存储节点上,由于信息的分布式存储,当用户进行信息查询时,查询效率很低,因此也需要构建分布式时空索引,提高查询效率。当各传感器采集到信息之后,采用分布式时空查询处理系统将信息进行索引,从而提高查询效率。例如用户可以查询在某次物流运输过程中,某货物在指定的时间范围内经过哪些中转站。

6　结束语

随着越来越多的分布于各地的时空数据的积累,分布式时空管理技术是必然的有效管理手段。本文从如何有效进行分布式时空查询的角度出发,提出了基于对等计算的时空查询处理的机制,设计了相应的查询处理系统,重点描述了在这种环境的存储和索引机制、将来预测查询处理和历史查询处理过程,并进行相关实验且效果显著,最后基于所设计的系统讨论了应用实例。

参考文献:

[1] ABRAHAM T, RODDICK J F. Survey of spatio-temporal databases [J]. GeoInformatica,1999,3(1):61-99.

[2] THEODORIDIS Y, SELLIS T, PAPADOPOULOS A N, *et al*. Specifications for efficient indexing in spatiotemporal databases[C]//Proc of the 10th International Conference on Scientific and Statistical Database Management. 1998:123-132.

[3] NASCIMENTO M, SILVA J. Towards historical R-trees[C]//Proc of ACM Symposium on Applied Computing. New York: ACM Press, 1998:235-240.

[4] TAO Yu-fei, PAPADIAS D, SUN Ji-meng. The TPR*-Tree: an optimized spatio-temporal access method for predictive queries[C]// Proc of the International Conference on Very Large Data Bases (VLDB). 2003.

[5] STOICA I, MORRIS R, KARGER D, *et al*. Chord: a scalable peer-to-peer lookup service for internet applications[C]//Proc of the ACM SIGCOMM. 2001:149-160.

[6] FALOUSOS C, ROSEMAN S. Fractals for secondary key retrieval [C]//Proc of the 8th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. New York:ACM Press,1989: 247-252.

[7] PeerSim simulator project[EB/OL]. (2005-12-18) [2009-05-20]. Http://peersim. sourceforge. net.

[8] THEODORIDIS Y, SILVA J R O, NASCIMENTO M A. On the generation of spatiotemporal datasets[C]//Proc of the 6th International Symposium on Spatial Databases. 1999:147-164.

[9] International Telecommunication Union. The Internet of things: report of ITU[R]. Tunis:ITU,2005.

(上接第 1341 页)

[6] 孟仲伟,鲁宗相,宋靖雁. 中美电网的小世界拓扑模型比较分析[J]. 电力系统自动化,2004,28(15):21-29.

[7] 曹一家,陈晓刚,孙可. 基于复杂网络理论的大型电力系统脆弱线路辨识[J]. 电力自动化设备,2006,26(12):1-5.

[8] 陈晓刚,孙可,曹一家. 基于复杂网络理论的大电网结构脆弱性分析[J]. 电工技术学报,2007,22(10):138-144.

[9] 曹一家,王光增,包哲静,等. 一种复杂电力网络时空演化模型[J]. 电力自动化设备,2009,29(1):1-5.

[10] CAO Yi-jia,WANG Guang-zeng,JIANG Quan-yuan, *et al*. A neighborhood evolving network model[J]. Physics Letters A,2006,349(6):462-466.

[11] LI Xiang, CHEN Guan-rong. A local-world evolving network model [J]. Physica A: Statistical Mechanics and its Applications, 2003,328(1/2):274-286.

[12] GUO Wei-ping,LI Xiang,WANG Xiao-fan. Epidemics and immunization on Euclidean distance preferred small-world networks[J]. Physica A:Statistical Mechanics and its Applications,2007,380:684-690.

[13] BARRAT A,BARTHELEMY M,VESPIGNANI A. Weighted evolving networks: coupling topology and weight dynamics [J]. Physical Review Letters,2004,92(22):1-4.

[14] 王光增,曹一家,包哲静,等. 一种新型电力网络局域世界演化模型[J]. 物理学报,2009,58(6):3597-3602.

[15] BALLOU H R. 企业物流管理供应链的规划、组织与控制 [M]. 王晓东,等译. 北京:机械工业出版社,2002.