

文章编号: 1007-4619(2008)05-0831-06

空间信息网格中远程空间连接查询的优化

谢传节, 刘高焕, 高秉博, 盛文涛

(中国科学院 地理科学与资源研究所, 资源与环境信息系统国家重点实验室, 北京 100101)

摘 要: 随着空间信息网格的建设, 网格平台上管理的空间信息资源越来越丰富, 这促进了空间信息网格中空间数据分布式查询的应用需求, 而在分布式空间查询中, 空间连接查询操作往往成为性能的瓶颈。根据空间信息的特点, 通过利用网格计算资源来优化空间连接查询的执行。首先基于网格服务构建网格平台分布式空间数据查询软件结构, 通过设计远程空间连接执行服务利用网格平台中的计算资源; 根据空间信息的特点, 采用基于 Kd-Tree 空间分区并行连接的方法提高远程空间数据连接操作执行效率, 并给出了远程空间连接执行的查询代价模型; 然后根据连接代价模型设计了远程空间连接查询执行计划优化生成算法; 最后总结了本文工作并探讨了下一步研究方向。

关键词: 空间信息网格; 分布式空间查询优化; 远程空间连接查询; 并行查询

中图分类号: P208 **文献标识码:** A

1 引 言

“网格计算”概念的提出, 得到了高度的重视, 并迅速掀起了研究和应用的热潮。对网格的理解已经不局限于科学计算, 扩展到数据网格、服务网格、仪器网格、虚拟现实网格、信息网格和知识网格等方面^[1-3]。网格计算平台快速应用到大气模拟, 资源环境信息管理等应用领域^[4-5]。

网格平台同空间信息的结合, 得到了 GIS 领域的关注和重视, 一些研究分析了空间分析同网格计算结合的可行性^[6], 在运用网格平台上的计算资源来提高空间分析效率方面进行了一些探索性研究, 并在遥感信息和网格计算结合方面取得了一些进展^[7-9]。在应用空间信息, 国际上出现了一些网格计算技术为基础的空间信息共享和应用平台^[4, 10], 如 Earth System Grid。通过网格连接的空间信息资源越来越丰富, 这也推动了网格平台上空间数据集成的应用需求。

网格平台上的信息集成问题一直是网格计算领域一个关键问题, 与此相关的研究, 如网格平台上数据资源的访问、网格平台上数据的分布式查询、数据缓存的运用、网格资源在数据集成中的运

用等方面研究大量涌现^[11-13]。网格平台信息集成研究中, 分布式数据查询技术受到了特别的关注, 如建立在 OGSA-DAI 基础上的 OGSA-DQP 就已网格服务的方式实现了分布式的数据查询^[14], 另外针对网格资源动态变化的特点, 出现了一些结合网格特点的分布式数据查询优化算法^[15]。

由于空间信息特殊性, 网格平台上空间信息的集成除了要汲取网格平台上一般信息集成的研究成果外, 更需要针对空间信息的特点, 研究和开发专用的空间信息集成技术。本文针对空间信息的特点, 研究了网格平台上空间信息联合查询中的一个关键问题, 即分布式空间数据的空间连接查询优化问题。

2 基于网格服务的分布式空间数据查询软件结构

当前网格平台上管理的空间信息资源越来越丰富, 这促进了分布式空间数据网格联合查询的应用需求。在网格平台上不同网格节点中空间数据的联合查询中, 空间连接查询操作是常见的功能, 同时也是分布式空间信息查询中计算需求最大的功能, 网格平台丰富的计算资源为优化分布式空间

收稿日期: 2007-07-28; 修订日期: 2008-03-10

基金项目: 国家高技术研究发展计划项目 (编号: 2007AA12Z203) 和中国科学院创新研究领域前沿项目资金的支持。

作者简介: 谢传节 (1971—), 男, 副研究员, 硕士生导师, 2000 年毕业于中国科学院地理科学与资源研究所, 获博士学位。主要研究方向为分布式地理信息系统、网络三维地理信息系统。E-mail: xiejc@lreis.ac.cn

数据查询连接操作,提高分布式空间数据联合查询的效率提供了良好的基础。

分布式空间数据查询软件由 3 种类型的网格服务构成,即分布式空间数据查询服务、空间数据

服务和空间数据连接执行服务,这 3 种类型服务相互协作共同完整分布式空间数据查询的优化和执行。图 1 是分布式空间数据查询系统中不同类型服务之间的协作关系图。

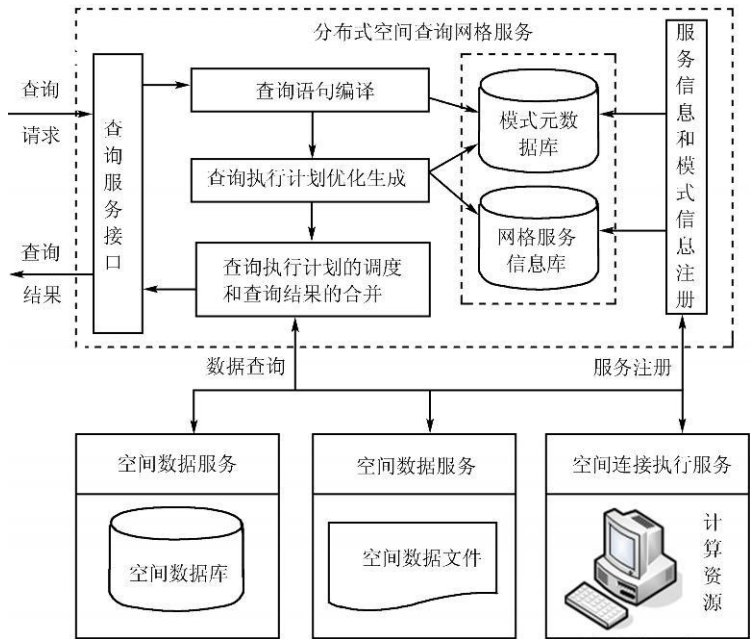


图 1 分布式空间数据查询系统中不同类型服务之间协作关系图

Fig. 1 The cooperation of the different kinds of services in the distributed spatial query system

分布式空间数据查询服务是基于网格的分布式空间数据查询系统的主体部分,分布式查询服务具有 3 个方面的功能:即网格服务注册功能、模式管理功能和查询功能。网格服务注册功能负责登记可用的空间数据服务和空间连接执行服务;模式管理服务通过访问空间数据服务获取模式信息,并可接受用户的编辑,定义组合的全局模式;查询功能接受外部的查询语句,通过查询解析和编译生成查询语句语法树,再结合服务信息、模式信息和网格资源信息进行查询优化,形成优化的查询执行计划,最后由查询计划调度系统将子查询调度到不同的数据服务和连接计算服务中,将查询的结果进行合并处理并返回查询结果。

数据服务封装了不同结构、不同形式存储的本地空间数据,提供本地数据的标准化查询能力。数据服务应能独立自主运行,能自主响应分布式查询服务提交的子查询,也能够响应其他系统的查询请求;数据服务应具有向外部应用描述其服务能力和所支持的数据模式信息,这些信息将登记到分布式查询服务中,为全局模式的构建和查询的优化提供必要的信息。数据服务的建设要尽可能遵从已有

的标准,特别是 OpenGIS 系列标准。

空间连接执行服务用来执行分布式数据查询中,不同数据服务间空间数据远程连接操作。分布在不同网格节点数据服务中的空间数据连接是一个对计算资源和网络传输资源有很高要求的操作,通过使用共享的网格计算节点来执行空间连接操作,可以利用网格计算平台中的计算资源,提高分布式空间数据执行的效率。

3 基于 Kd-Tree 分区空间连接并行执行及其代价模型

当参与远程连接的空间数据集数据量较大时,空间连接操作将成为整个空间数据查询的性能瓶颈,这需要考虑同时利用多个空间连接执行服务,采用并行的方式来执行空间连接操作,可采用空间分区并行执行的方法来提高空间连接操作的效率。

空间分区的方法一般可采用两种,即规则的网格的简单空间分区方法和空间区域递归划分的复杂方法。简单空间分区的缺点是当数据分布不均匀时将导致个别某些子连接操作涉及的对象数量

过多,降低连接执行的效率。

比较理想的是根据数据的分布情况,通过控制在子区域中对象的数量最大值进行递归空间分割,为了提高递归空间分区效率,可将该操作设计为一个数据服务支持的基本功能。递归空间分区可采用类似 Kd-Tree 的方法进行分区,通过控制每个 Kd-Tree 的叶结点包含空间对象的最大数量达到控制并行化程度的目的。图 2 是当假定空间区域包含空间对象数不超过 2 时按照递归空间分区图。

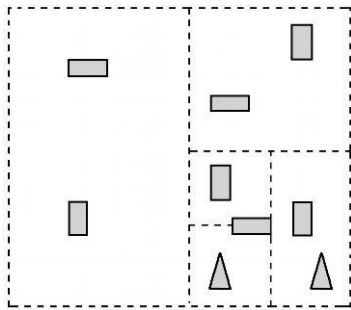


图 2 空间子区域包含空间对象数不超过 2 时的
空间分区图

Fig.2 The spatial partition results of the Kd-Tree on the control of the maximum objects a leaf node contained no more than two

在对并行化后的子区域的连接操作执行过程中,由于有些空间对象可能同时与多个子区域相交,所以不同子区域中可能存在冗余的空间对象,这需要在子查询结果合并时予以消除。

在查询执行中,远程空间连接操作是否要采用空间分区并行方法,以及分区参数的确定需要以查询执行的代价为依据。分布式空间数据连接查询代价主要由两部分构成,一部分是查询执行代价,另一部分是数据网络传输代价。

分布式空间连接操作是通过空间连接计算服务完成,该服务将参与空间连接的数据集读取到空间连接服务中完成空间连接操作。设有分布在不同数据服务节点的空间数据集 R 、 S 和空间连接操作服务节点 JS $|R|$ 为数据集 R 的元组数, $|S|$ 为数据集 S 的元组数, $\bar{r}$ 、 $\bar{s}$ 分别为数据集 R 、 S 中元组的平均大小, C_{opr}^{JS} 为单个连接操作的在 JS 上的执行代价, C_{comm}^{R-JS} 为从数据集 R 所在网格节点向 JS 所在网格节点的单位数据传输代价, C_{comm}^{S-JS} 为从数据集 S 所在网格节点向 JS 所在网格节点的单位数据传输代价。此时一般的连接操作的过程为:

- (1) 将 R 传输到 JS
- (2) 将 S 传输到 JS
- (3) 在 JS 上执行空间连接操作。

此时 R 、 S 上的分布式连接总代价为:

$$T = (|R| \times |S|) \times C_{opr}^{JS} + (|R| \times \bar{r}) \times C_{comm}^{R-JS} + (|S| \times \bar{s}) \times C_{comm}^{S-JS} \quad (1)$$

在经过区域划分后,参与空间连接的数据集 R 和 S 采用相同的空间划分, R 可以表达为 $R = \sum R_i$, R_i 为空间区域 i 的对应的子数据集, S 可以表达为 $S = \sum S_i$, S_i 为空间区域 i 的对应的子数据集,空间区域 i 中 R 和 S 的连接操作在连接操作服务节点 JS_i 上执行,则空间区域 i 中 R 和 S 的空间数据连接的代价为 T_{ii} ,则在一般空间连接情况下, CT_{ii} 为:

$$T_{ii} = (|R_i| \times |S_i|) \times C_{opr}^{JS_i} + (|R_i| \times \bar{r}) \times C_{comm}^{R-JS_i} + (|S_i| \times \bar{s}) \times C_{comm}^{S-JS_i} \quad (2)$$

整个连接查询的代价可以取最大的分区连接的代价值:

$$T = \sum CT_{ii} \text{ 其中 } i \text{ 为分区编号。} \quad (3)$$

4 远程空间连接查询执行计划的优化生成算法

远程空间连接的执行计划规定了远程空间连接执行的方式,包括空间分区并行的确定、连接查询执行服务的选择和分区查询任务向连接执行服务的分配。查询执行计划优化生成的依据是查询的代价模型公式。

查询代价公式计算过程中,关键是代价模型中参数的确定,包括数据集的相关信息,如数据集元组数、元组平均大小;连接服务的相关信息,如单个连接操作执行时间;单位数据的传输代价是网格平台网络的属性,由网格资源管理器维护,如 Globus 平台上的 network weather service 可提供网络能力监控功能。在分布式查询网格服务软件体系中要获取这些参数,需要在各种网格服务设计时考虑到这些参数的输出和管理。分布式查询网格服务软件以 WSRF 体系为依托,上述参数可以作为服务资源的属性而存在。数据集的相关信息可以作为空间数据网格服务的属性,并且注册到分布式查询软件的服务信息管理器中。分区并行时分区范围确定则需要进一步通过扩展网格服务的功能才能得到。空间连接操作执行时间可以作为空间连接执行服

务资源的质量属性、单位数据的传输代价可以作为网格资源的质量属性,这些质量属性可以在连接服务执行过程中动态更新。下面算法 1 是基本的远程空间连接执行计划的优化生成算法:

4.1 算法 1 基本的远程空间连接执行计划的优化生成算法

Step 1: 初始化
设定一个阈值 α 表示当选定一个新查询计划时,该查询计划与上一个选定的查询计划相比较,在查询性能上能够提升比例的下限;
分析空间连接查询语句,获取参与空间连接查询语句的空间数据网格服务;与网格资源管理设施联系,获取上述空间数据网格服务资源所支持的空间数据集信息,并分析和比较查询语句中相关数据集的查询字段使用是否正确;
通过联结空间数据查询软件网格服务管理软件,查询当前可用的空间连接执行网格服务;
从可用的网格连接服务中选择一个空间连接网格服务,讲该服务加入到已用服务堆栈中,并从网格资源管理设施联系获取该服务质量属性和相关网络资源质量属性,在根据公式 (1) 计算查询代价。

Step 2: 分区并行方案确定
重复执行 Step2。
如果所有的空间连接网格服务已进入已用服务堆栈,则当前连接执行方案被确定,并跳转进入 Step 3; 否则从未选入已用服务堆栈中的空间连接网格服务中选取一个连接服务,并且总是将堆栈中的服务数量作为分区并行式的分区数量,由此确定分区中对象数量,并从网格资源管理设施联系获取参与当前连接查询服务质量属性和相关网络资源质量属性,按照公式 (2)、(3) 计算查询代价。

计算当前查询计划与前一个查询计划相比所带来的性能提升的比例,如果超过阈值 α 则选定新的查询计划,并重复执行 Step 2。

Until 性能提升比例超过阈值。
Step 3: 查询计划生成
如果当前已用服务堆栈为空,则查询计划生成失败,退出算法执行;

如果服务堆栈中有 1 个可用服务,则按照单节点执行的方式编制空间连接查询执行计划;

如果服务堆栈中有多个服务可用,则根据服务数量计算 Kd-Tree 分区时每个分区最大对象数,选

择参与连接空间数据集中空间对象数最多的数据服务,调用该服务扩展的 Kd-Tree 分区功能,获取分区方案;分区数可能超过可用连接服务数,此时可合理合并部分分区,让分区数维持与可用连接服务数相等,然后根据分区方案,按照多节点并行执行的方式编制空间连接查询执行计划。

End 算法输出远程空间连接查询执行计划。
这里需要注意算法 1 中,对于空间数据网格服务,由于其网格服务资源属性是稳定的,只需要在初始化时与网格资源管理服务一次通信;而空间连接查询服务质量属性和网络资源质量属性处在变化中,需在每次计算代价值时与网格资源管理服务通信,总是需要获取这些网格资源质量属性的当前值。

一次分布式空间数据查询语句可能有多个分布式空间连接操作,所以算法 1 需要被调用多次。所以在完整的分布式空间数据查询优化算法中,当前远程空间连接操作执行计划生成后,需要再次访问查询语句,找出下一个分布式空间连接操作,只有当查询语句中所有分布式连接查询优化执行结束时整个的查询优化工作才能完成。下面算法 2 是完整的分布式空间数据查询优化算法。

4.2 算法 2 完整的分布式空间数据查询优化算法

Step 1: 初始化
解析用户查询语句,构建查询语法树,标识分布式空间数据连接查询操作。
Step 2: 分布式空间连接查询优化
重复执行 Step 2。
从查询计划中选择分布式空间数据连接操作。
针对选定的分布式空间数据连接,执行算法 1,获取当前分布式空间数据连接查询执行计划。
Until 所有空间连接操作执行计划均已生产。
Step 3: 分布式空间数据查询执行计划生成。
按照查询语法树组合分布式空间数据连接执行计划,形成完整的分布式空间数据查询执行计划。
End 输出完整的分布式空间数据查询执行计划

5 结 论

随着空间信息网格的建设,网格平台上管理的空间信息资源越来越丰富,这促进了分布式空间数据网格分布式查询的应用需求,而在分布式查询

中,空间连接查询操作是非常常见的功能,同时也是分布式空间信息查询中计算需求最大的分析功能,需要充分利用网格平台丰富的计算资源,优化分布式空间数据查询连接操作执行,从而提高分布式空间数据查询的效率。

文中基于网格服务构建网格平台分布式空间数据查询软件结构,并通过设置远程空间连接执行服务利用网格平台中的计算资源;接着根据空间信息的特点,采用基于 Kd-Tree 空间分割并行连接的方法提高远程空间数据查询连接操作执行效率,并给出了远程空间连接执行的查询代价模型;然后根据连接代价模型设计了远程空间连接查询执行计划生成算法。

网格是一个动态变化的环境,同时也是一个有良好管理能力的环境。一方面通过网格计算平台提供的各种管理设施可以构建分布式空间数据查询执行的监控软件;另一方面以监控软件获取的执行状态信息作为反馈式输入,可以进一步根据查询实际执行情况,对查询执行计划进行再优化,从而提高远程空间连接优化执行的自适应能力。

参考文献 (References)

- [1] Chen B, Xiao N, Liu B. A Distributed Data Server in Grid Environment[A]. Lecture Notes in Computer Sciences[C]. GCC 2004 Workshops 2004, **3252**, 775—782.
- [2] Li Y, S Yang S W, Jiang J L, et al. Build Grid-Enabled Large-Scale collaboration Environment in e-Learning Grid[J]. Expert Systems with Applications 2006, **31**, 742—754.
- [3] Foster I, Kesselman C, Nick J M, et al. Grid Services for Distributed System Integration [J]. IEEE Computer 2002, **35** (6), 37—46.
- [4] Ann Chervenak, Ewa Deelman, Carl Kesselman, et al. High-Performance Remote Access to Climate Simulation Data: A Challenge Problem for Data Grid Technologies [J]. Parallel Computing 2003, **29**, 1335—1356.
- [5] David Bemholt, Shishir Bharathi, David Brown, et al. The Earth System Grid: Supporting the Next Generation of Climate Modeling Research[A]. Proceedings of the IEEE [C]. 2005, **93**(3), 485—495.
- [6] Hu Y C, Xue Y, Wang J Q, et al. Feasibility Study of Geo-spatial Analysis Using Grid Computing[A]. Lecture Notes in Computer Sciences[C]. ICCS 2004, **3039**, 956—963.
- [7] Xue Y, Wang J, Wang Y G, et al. Preliminary Study of Grid Computing for Remotely Sense Information [J]. International Journal of Remote Sensing 2005, **26**(16), 3613—3630.
- [8] Xue Y, Wang Y G, Wang J Q, et al. High Throughput Computing for Spatial Information Processing (HT-SIP) System on Grid Platform[A]. Lecture Notes in Computer Sciences[C]. EGC 2005, **3470**, 40—49.
- [9] Shen Z F, Luo J C, Zhou C H, et al. System Design and Implementation of Digital Image Processing Using Computational Grids[J]. Computers & Geosciences 2005, **31**, 619—630.
- [10] Chu K D, Di L P, Thomson Peter. Introduction of Grid Computing Application Projects at the NASA Earth Science Technology Office[A]. Lecture Notes in Computer Sciences[C]. GPC 2006, **3947**, 289—298.
- [11] Sivanmakrishnan Narayanan, Um it Catalyurek, Tahsin Kuro, et al. Applying Database Support for Large Scale Data Driven Science in Distributed Environments[A]. Fourth International Workshop on Grid Computing[C]. 2003.
- [12] Alexander Wohrer, Peter Brezany, A M in Tjoa. Novel Mediator Architectures for Grid Information Systems [J]. Future Generation Computer Systems 2005, **21**, 107—114.
- [13] Jim Smith, Anastasios Gounaris, Paul Watson, et al. Distributed Query Processing on the Grid[A]. Lecture Notes in Computer Science[C]. GRID 2002, **2536**, 279—290.
- [14] M. Nedim Alpdemir, Arijit Mukherjee, Anastasios Gounaris, et al. Using OGSA-DQP to Support Scientific Applications for the Grid[A]. Lecture Notes in Computer Science[C]. SAG, 2004, **3458**, 13—24.
- [15] Anastasios Gounaris, Jim Smith, Norman W Paton, et al. Adapting to Changing Resource Performance in Grid Query Processing[A]. Lecture Notes in Computer Science[C]. In Int Workshop on Data Management in Grids volume 3836 of LNCS Springer 2005, 30—44.

The Optimization of Remote Spatial Join Queries on Spatial Information Grid

XIE Chuan-jie LIU Gao-huan GAO Bing-bo SHENG Wen-tao

(State Key Laboratory of Resources and Environment Information System, Institute of Geographical Sciences & Natural Resources Research, CAS, Beijing 100101, China)

Abstract With the application of the Spatial Information Grid (SIG), the spatial information managed by SIG being more and more abundant. The abundant spatial information requires the better application of distributed spatial information query across SIG. However, the remote spatial join queries are always the bottleneck in the distributed spatial information query. Based on this observation, in this paper, the spatial join queries are optimized by taking full advantage of the grid computing resources according to the characteristics of spatial information.

At first, the software architecture for distributed spatial query is designed based on the different grid services. The distributed spatial data query software architecture is composed of three different kinds of grid services, namely: Distributed Spatial Data Query Grid Service (DSDQGS), Spatial Data Grid Service (SDGS) and Remote Spatial Join Query Grid Service (RSJQGS). These three kinds of grid services cooperate to implement the optimization and execution of the distributed spatial data query. In the architecture, the grid computing resources are utilized by the remote spatial join queries execution grid services.

Secondly, the partitioned parallel spatial join queries are implemented by the Kd-Tree spatial partition scheme. In the scheme, an original spatial query is rewrite into several sub-spatial queries bounded by sub-regions of the Kd-Tree nodes, which can be run concurrently; therefore, the performance of the remote spatial join queries is improved. The cost model for the partitioned parallel spatial join queries is also presented in paper. The cost of the remote spatial join query involves two parts: the computing cost of the join operation and communication cost of the spatial data.

Thirdly, the optimization algorithm for the query planed to generate the remote spatial join queries is designed according to the cost model. The remote spatial join query plan prescribes the way the spatial join query execution, including the scheme for partitioned parallel spatial join query, SDGSs participated in the join query, and assignments of the partitioned parallel spatial join query tasks to RSJQGSs. The cost is benchmark for the remote spatial join query plan. The parameters utilized in the optimization algorithm are managed as properties of the WSRF. The full optimization algorithm is finished when all identified spatial join operators are processed. At last, the future research directions for the optimization of spatial distributed query on SIG are discussed.

Key words spatial information grid; remote spatial join queries; distributed spatial query optimization; parallel query