

面向多源异构数据源的实际范围索引树索引方法

吴润泽¹, 蔡永涛¹, 陈文伟¹, 陈文刚², 王一蓉³

(1. 华北电力大学电气与电子工程学院, 北京市 102206; 2. 国网山西省电力公司晋城供电公司, 山西省晋城市 048000;
3. 北京国电网络技术有限公司, 北京市 100070)

摘要:在发电、输配电、新能源接入及新型能源消费等各个环节产生的大数据需要高效、可靠的数据存储和管理。文中以能源互联网作为虚拟应用场景,提出了集群式多源异构数据存储方案。为满足集群存储中高效查询需求,构建了实际范围索引树(AR-tree)分层索引模型,并提出了面向多源异构数据源的 AR-tree 索引方法,该方法针对在双层索引模型的基础上,以局部数据索引的实际索引范围为对象,建立全局索引。同时,对所提 AR-tree 索引方法操作开销进行了综合分析。最后,仿真结果表明 AR-tree 索引方法能提高查询命中率并提升查询等操作效率。

关键词:能源互联; 智能电网; 大数据; 索引; 查询; 数据存储; 集群

0 引言

随着电力系统的深入发展,采用云计算平台处理电力系统各个环节的海量数据已达成广泛的共识。电网运行数据^[1-2]、新能源并网数据、智能用电数据等海量数据存储可以通过云计算平台分布式系统进行扩展^[3],但数据管理过程中多样、频繁的数据交互访问离不开高效索引和查询技术的支撑。

索引和查询是数据管理的重要入口,传统的单层关系型数据库索引,如 B 树索引^[4-5]、B+ 树索引^[6-7]、R 树索引^[8-9]、位图索引在可扩展性上难以满足数据快速增长的需求,而以 Hadoop 的 Hbase 为代表的列式数据库采用的 LSM 树索引^[10]能减少数据频繁的插入、修改等操作带来的磁盘读写次数,但是只能按照行键组织数据。目前,海量数据的索引研究领域包括:以 Hadoop 为基础的索引方法,处理效率高,但创建索引的代价太大;以 RBI 为代表的位图索引,属性阈值划分复杂,影响了该方法的可用性。双层索引作为解决分布式文件系统的数据索引问题的一个研究热点,国内外已经取得了不少研究成果,文献[11]研究了一种对等网络和 B+ 树相结合的索引方案。该方案在子节点上采用 B+ 树索引,并通过选举算法将每棵局部 B+ 树上的部分节点通过 BATON 网络分发到全局节点,该方法维护复杂,且局部节点的分发不能代表整棵局部树,查询

命中率低,并且需要一定的网络开销。文献[12-13]采用了和文献[11]类似的策略,并针对多维查询做出了改进,采用对等网络对节点进行覆盖,存在一定的网络通信开销,对于数据的频繁插入,维护代价过大。此外,主流的大数据管理系统采用的是集中式二级主从结构或是类主从结构,在这种结构上构建基于对等网络的索引会对原系统可靠性及稳定性造成一定影响。

双层索引是避免全表扫描,提高能源数据管理及查询效率的有效方式。在双层索引结构中,局部数据节点与全局索引之间并非一一映射,而是通过全局索引定位到具体的数据节点,再通过数据节点的局部索引操作数据。本文在分析能源互联网多源大数据构成的基础上,针对虚拟应用场景的多源异构数据操作效率低下的问题,设计了一种实际范围索引树(actual range tree, AR-tree)的索引结构,并给出了 AR-tree 索引的架构及算法,该算法在局部节点分别建立索引的基础上对局部索引的实际范围(AR)值建立全局索引。实验结果表明,AR-tree 索引有效提高了查询等数据操作的效率。

1 数据构成与存储

能源互联网是能源技术与信息技术深度耦合的能源利用体系,更加强调多种能源形式之间的综合互补利用^[14-15]。能源互联网本质上是在传统能源网络的基础上以能源互联机制为核心,通过信息通信技术、能源互联技术整合出一个能源共享的能源消费机制。其架构如图 1 所示。

收稿日期: 2015-08-29; 修回日期: 2016-01-11。

上网日期: 2016-03-24。

国家自然科学基金资助项目(51507063)。

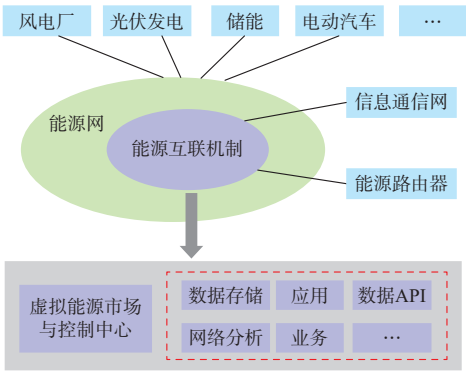


图 1 能源互联网虚拟架构
Fig.1 Virtual architecture of Energy Internet

由图 1 可知,能源互联所带来的的是虚拟能源市场的发展,能源互联网所产生的数据在支持能源互联网自身的运行及业务的同时,也向外部提供数据应用编程接口(API)。因此,要求能源互联数据存储满足异构融合、快速读写的要求。在能源互联网中,能源生产、调度、传输及消费中,产生和涉及的数据种类繁多,如网络故障信息、历史负荷数据、设备状态数据、天气数据等,构成了能源互联网多源大数据。为满足这些数据的存储需求,本文提出了集群存储方案,利用集群保证存储侧的负载均衡、高速 I/O 及可扩展性。图 2 显示了能源互联网大数据的集群存储方案。

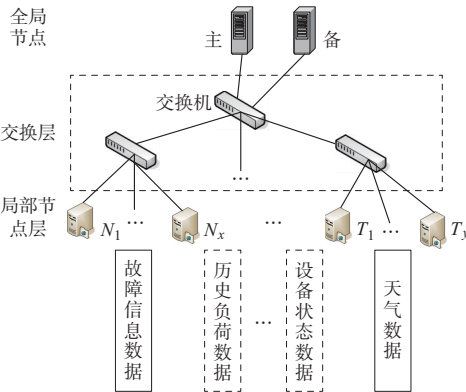


图 2 能源互联网虚拟场景下的多源异构数据存储方案
Fig.2 Storage scheme of multi-source heterogeneous data in virtual scene of Energy Internet

集群存储划分为全局节点层、交换层和局部节点层 3 个层次。全局节点层由主节点和备份节点构成,对上为用户提供透明的数据管理服务,对下管理存储子节点;交换层由一系列的路由器和交换机构成,为全局节点和局部节点提供通信服务;局部节点层采用的是独立的主机,互相之间不共享计算和存储资源,因此集群具有可扩展性。局部节点根据不

同的存储任务划分成不同组,每组分别存储不同的能源互联网数据信息,例如图 2 中的 N_1 至 N_x 存储故障信息数据, T_1 至 T_y 存储天气数据。

2 索引算法描述与设计

能源及其生产消费的特点决定了能源互联网数据管理的周转周期。以故障信息数据为例,当电网元件出现故障,引起能源供应服务的中断,则要求系统迅速诊断故障原因,给出处理方案,因此要求系统对于故障信息库的查询作业能够快速完成。本文采用 AR-tree 索引结构组织数据。AR-tree 采用的是分层设计思想,分为全局索引层和局部索引层。局部索引在局部节点上组织存储在本地的数据,并维护本地的 AR 值,全局索引对各个局部节点的 AR 值再次建立索引树,与局部索引树一起,构成 AR-tree 索引。AR-tree 整体分层结构如图 3 所示。

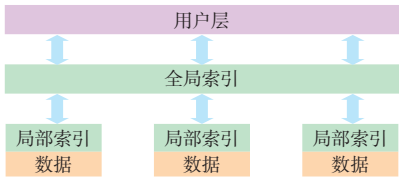


图 3 AR-tree 分层结构图
Fig.3 Layered structure of AR-tree

用户通过用户层 UI 界面或者用户指令送达操作指令;全局索引维护局部索引对上接收用户操作指令,对下维护局部索引树的 AR 值;局部索引负责组织本地节点上的数据,并维护自身的 AR 值。用户通过访问接口先查询第一层全局索引,以定位具体的局部节点,然后通过部署在局部节点上的局部索引检索数据。全局索引和局部索引的选择灵活多样,可以根据需要采用不同的组合方式,如都采用 B+ 树,或全局采用 B 树,局部采用 R 树等。局部索引与全局索引均采用 B 树组织节点,局部索引内节点关键字数据记录为主键,而全局索引树的内节点关键字为若干局部索引的存储范围及指向对应局部树的指针,表 1 为全局索引树与局部索引树节点对比。

表 1 局部索引与全局索引节点对比
Table 1 Comparison between local and global indexing nodes

局部索引节点	全局索引节点
KeyList	AR_KeyList
Childrenlist	Childrenlist
Isleaf	Isleaf
—	LocalTree
Memberfunctions	Memberfunctions

KeyList 为局部节点关键字;AR_KeyList 为全局节点关键字;Childrenlist 为当前节点所带有的孩子节点;Isleaf 标记当前节点是否是叶子节点;LocalTree 存放节点索引树信息。在局部索引中,索引树不光要维护索引本身,同时要维护每棵树的 AR,图 4 为 AR-tree 索引组织示意图。

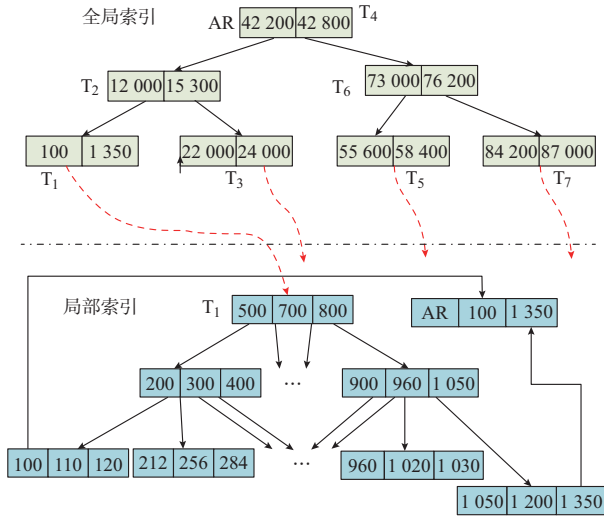


图 4 AR-tree 索引组织示意图

Fig.4 Schematic diagram of AR-tree structure

T 组节点为存储集群上的存储某类数据的一组节点, $T_i (i=1,2,\dots,7)$ 。每个局部索引的 AR 值都会上传到全局索引所在的节点,本地节点 T_1 将 T_1 的 AR 值 [100, 1 350] 上传至全局索引,用来构成全局索引中的关键字。因而全局索引节点关键字是一个包含一对数据关键字范围以及局部索引定位指针的复合数据类型。

3 算法流程与步骤

3.1 AR-tree 查询算法流程

对于多源异构属性的大数据存储操作而言,查询是最基础的操作,也是 AR-tree 索引方法的核心。算法描述中涉及的变量及其含义如下: key 是查询关键字, Ftree 是 AR-tree 的第一层索引树, 本文采用的是传统的 B 树索引, AR 是 Ftree 节点所包含的关键字对象属性之一, Btree 是 AR-tree 索引结构的第二层局部索引树。算法步骤如下。

1) 先查询 Ftree, 如果 Ftree 中的节点关键字中不存在一个 AR 值 [min, max] 使得 key 落在范围 [min, max] 中, 查询结束。

2) 如果 Ftree 中的节点关键字中存在一个这样的 AR, 则查询该关键字指向的 Btree。

3) 如果 Btree 节点中存在 Key, 则查询结束。

4) 算法结束。

由于 AR-tree 创建时数据已按照主键排序, 则至多存在一个满足条件的 AR。对于范围查询, 如查询范围 (key1, key2), 需要将范围的最大、最小值关键字 key2 和 key1 分别取出, 并检索两个关键字, 最终将两个关键字之间的所有关键字记录返回。

3.2 AR-tree 索引插入算法流程

插入操作是在查询的基础上进行的, 如果查询到的关键字已经存在, 则不插入, 反之, 则按照 B 树的向上分裂原则进行插入。插入算法描述如下。

1) 查询 key 是否已经存在, 如存在, 插入结束。

2) 查询包含 key 的 AR, 如果存在, 将 key 插入响应局部树, 插入结束。

3) 如果 key 落在两个 AR 之间, 判断之后插入, 插入结束。

4) 算法结束。

B 树是向上增长, 且插入之前均要判断节点内关键字是否已满, 如果已满则需要节点分裂, 从而保持树的平衡。插入操作中的一个特殊情况是, 待插入的关键字不落在任何子树已存在的范围, 如图 4 中, 一个待插入的关键字 $key=10\ 000$, 落在了 T_1 的 AR 和 T_2 的 AR 之间, 针对此类情况, 本文中采取的策略是随机选取一个局部子树进行插入, 并且一旦多次连续插入某一棵子树, 则下次强制插入另外的一棵子树。

3.3 AR-tree 操作开销分析

对于 AR-tree 而言, 其操作开销包括两部分: 全局索引和局部索引。假设 N 条数据存入单一 B 树, 允许最小度数为 $t \geq 2$, 该树的高度 h_1 满足:

$$h_1 \leq \log_t \left(\frac{N+1}{2} \right) \quad (1)$$

同理, 假设在 AR-tree 中, 有 k 个局部节点, 两层索引树中, 最小度数同设为 t 。那么局部索引树和全局索引树的平均高度 h_2 和 h_Δ 分别满足:

$$h_2 \leq \log_t \left(\frac{N+k}{2k} \right) \quad (2)$$

$$h_\Delta = \log_t \left(\frac{k+1}{2} \right) \quad (3)$$

按照树的高度最大计, 并由 $N \gg k$ 及 $\log$ 函数的性质, 不难得出:

$$h_1 - (h_2 + h_\Delta) = \log_t \left(1 + \frac{N+1-k}{N+k+1} \right) > 0 \quad (4)$$

由式 (4) 得出结论, 在查询操作中, AR-tree 的对比次数小于 B 树, 即查询的时间开销更小。同理可得, 对于 AR-tree 的其他操作时间开销也更小。

4 实验与结果分析

B 树是一种平衡的多路查找树,在文件索引和数据索引中被广泛采用,一棵 m 阶 B 树至多有 m 棵子树。本文的 AR-tree 采用 B 树索引作为全局索引和局部索引,因此选择 B 树索引作为对比算法。本文从查询、插入、删除等操作内容方面测试 AR-tree。实验中 AR-tree 局部树存储整型关键字,实验操作数据量级如表 2 所示。

表 2 试验操作数据量级描述
Table 2 Data magnitude description of experimental operations

操作	数据量级
查询	10^7
插入	10^6
删除	10^6

实验中,每个操作在量级上分别重复 9 次或 10 次实验,每次实验数据量逐渐增加。实验中的数据随机产生,由于对比操作的数量级一致,因此随机数产生过程并不影响实验对比效果。

所有实验图显示时间取实际运行时间(单位为 ms)的 1/100,即图中的 10 表示实际运行时间为 1 000 ms。度数是每个节点能容纳的子树个数。查询实验中,将 AR-tree 的度数及 B 树度数均设置为 2,子树数目 k 设为 100。

从图 5 可知,AR-tree 的查询响应时间几乎是 B 树的 1/3,当查询请求数为 9×10^7 时,AR-tree 的响应时间是 100 个时间单位,而 B 树的响应时间是 345 个时间单位。由 3.3 节推导可知 $N=10^7, t=2, k=100$ 时,查询操作中 AR-tree 平均比较次数小于 B 树。AR-tree 查询性能优越的原因在于两点:一是 AR 值的设置和维护使得查询命中效率提高,有效减少了不存在关键字的无效查询,从而总体上降低了查询请求的响应时间;二是双层索引的设置使得局部子树的规模小于 B 树。

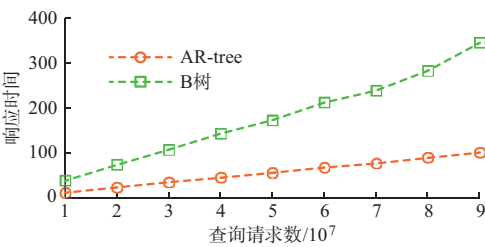


图 5 AR-tree 和 B 树查询操作响应时间对比
Fig.5 Comparison of query cost between AR-tree and B-tree

插入和删除的请求数量级均为 10^6 ,度数设置

同查询操作一致,实验结果对比见图 6 和图 7。从实验结果可见,AR-tree 的插入和删除操作响应时间基本与请求数成正比。B 树的插入响应时间前期也与插入请求数成正比,但随着插入请求数增大,其响应时间增长率逐渐变大,原因在于当索引树的规模变大时,插入操作产生的节点分裂增多,造成索引树调整开销增大。AR-tree 采用的是双层索引模型,当插入索引树规模增大时,每棵子树的平均增长规模小于 B 树,主要原因在于这两个操作的前提均为查询操作,而 AR-tree 查询操作的效率比 B 树更加优越。图 8 给出了全局索引度数为 2,3,4 时,查询响应时间的变化。

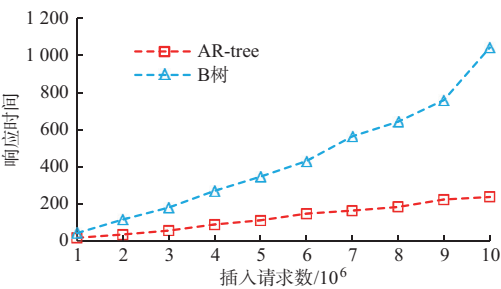


图 6 AR-tree 和 B 树插入操作响应时间对比
Fig.6 Comparison of insert cost between AR-tree and B-tree

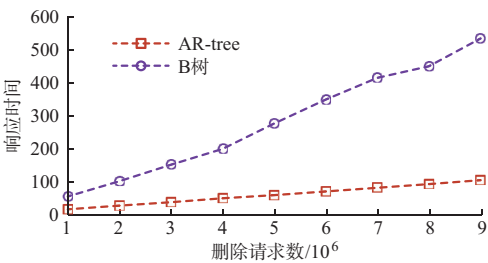


图 7 AR-tree 和 B 树删除操作响应时间对比
Fig.7 Comparison of deletion cost between AR-tree and B-tree

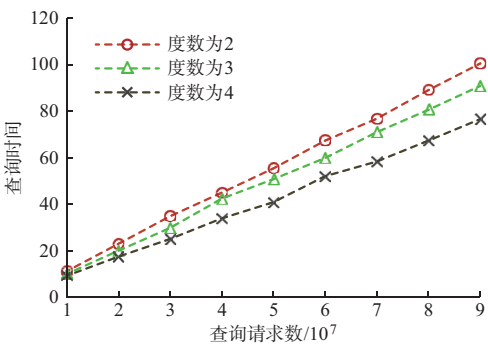


图 8 不同度数查询时间对比
Fig.8 Comparison of query cost among different degrees

对于 AR-tree 而言,当局部子树度数不变,而改变全局索引树的度数,其操作性能也会受到影响。由图 8 可知,度数增大,全局索引的深度越小,操作的响应时间响应变短。事实上,节点分裂和合并开销限制了索引度数设置的上限。

5 结语

在不同互联的能源体系中必然存在更多的数据交互,提升数据索引效率能有效提高数据操作效率并节约时间成本。本文以能源互联网作为虚拟应用场景,在其多源、异构数据特征基础上,提出了面向集群式多源异构数据存储的分层索引模型和 AR-tree 索引方法。而随着能源互联网的深入发展,能源的多样化则带来更多的数据频繁交互,传统索引将扩展为分布式集群索引,将涉及更大规模的数据频繁操作,因此 AR-tree 索引方法将进一步针对物理集群进行改进,此外,本文所提方法需在索引边界问题上进一步研究。

参考文献

- [1] 曲朝阳,陈帅,杨帆,等.基于云计算技术的电力大数据预处理属性约简方法[J].电力系统自动化,2014,38(8):67-71.DOI: 10.7500/AEPS20130601001.
- QU Zhaoyang, CHEN Shuai, YANG Fan, et al. An attribute reducing method for electric power big data preprocessing based on cloud computing technology[J]. Automation of Electric Power Systems, 2014, 38(8): 67-71. DOI: 10.7500/AEPS20130601001.
- [2] 梅华威,米增强,吴广磊.基于 MapReduce 模型的间歇性能源海量数据处理技术[J].电力系统自动化,2014,38(15):76-80.DOI: 10.7500/AEPS20131215006.
- MEI Huawei, MI Zengqiang, WU Guanglei. Massive data processing of intermittent energy based on MapReduce model[J]. Automation of Electric Power Systems, 2014, 38(15): 76-80. DOI: 10.7500/AEPS20131215006.
- [3] 王德文.基于云计算的电力数据中心基础架构及其关键技术[J].电力系统自动化,2012,36(11):67-71.
- WANG Dewen. Basic framework and key technology for a new generation of data center in electric power corporation based on cloud computation[J]. Automation of Electric Power Systems, 2012, 36(11): 67-71.
- [4] FERRAGINA P, VENTURINI R. Compressed cache-oblivious string B-tree[J]. Lecture Notes in Computer Science, 2013, 8125: 469-480.
- [5] HU Jinglin, QI Deyu. An efficient index structure for range count query based on improved B-tree[J]. International Journal of U-and E-Service, Science and Technology, 2015, 8(4): 159-168.
- [6] JIN Peiquan, YANG Puyuan, YUE Lihua. Optimizing B+-tree for hybrid storage systems [J]. Distributed and Parallel Databases, 2015, 33(3): 449-475.
- [7] HAVASI F. An improved B+ tree for flash file systems[J]. Lecture Notes in Computer Science, 2011, 6543: 297-307.
- [8] DEKIHARA H, NAKAMURA Y. An extended technique for R-tree to manage multiple type objects [J]. Journal of Computational Methods in Sciences & Engineering, 2012, 12(1): 53-61.
- [9] WANG P, RAVISHANKAR C V. Secure and efficient range queries on outsourced databases using Rp-trees[C]// IEEE 29th International Conference on Data Engineering (ICDE), April 8-12, 2013, Brisbane, Australia: 314-325.
- [10] O'NEIL P, CHENG E, GAWLICK D, et al. The log-structured merge-tree (LSM-tree) [J]. Acta Informatica Medica, 1996, 33(4): 351-383.
- [11] WU Sai, JIANG Dawei, OOI B C, et al. Efficient B-tree based indexing for cloud data processing[J]. Proceedings of the VLDB Endowment, 2010, 3(1/2): 1207-1218.
- [12] WANG Jinbao, WU Sai, GAO Hong, et al. Indexing multi-dimensional data in a cloud system[C]// Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, June 6-10, 2010, Indianapolis, USA: 591-602.
- [13] AGUILERA M K, GOLAB W M, SHAH M A. A practical scalable distributed B-tree [J]. Proceedings of the VLDB Endowment, 2008, 1(1): 598-609.
- [14] 董朝阳,赵俊华,文福拴,等.从智能电网到能源互联网:基本概念与研究框架[J].电力系统自动化,2014,38(15):1-11.DOI: 10.7500/AEPS20140613007.
- DONG Zhaoyang, ZHAO Junhua, WEN Fushuan, et al. From smart grid to energy internet: basic concept and research framework[J]. Automation of Electric Power Systems, 2014, 38(15): 1-11. DOI: 10.7500/AEPS20140613007.
- [15] 黄仁乐,蒲天骄,刘克文,等.城市能源互联网功能体系及应用方案设计[J].电力系统自动化,2015,39(9):26-33.DOI: 10.7500/AEPS20141229010.
- HUANG Renle, PU Tianjiao, LIU Kewen, et al. Design of hierarchy and functions of regional energy internet and its demonstration applications[J]. Automation of Electric Power Systems, 2015, 39(9): 26-33. DOI: 10.7500/AEPS20141229010.

吴润泽(1975—),女,博士,副教授,主要研究方向:现代信息通信技术及在电力系统中的应用。E-mail: wurz@ncepu.edu.cn

蔡永涛(1989—),男,通信作者,硕士研究生,主要研究方向:智能电网大数据存储与索引。E-mail: cyt_025@126.com

陈文伟(1991—),女,硕士研究生,主要研究方向:数据挖掘及其在智能电网中的应用。E-mail: 572451718@qq.com

(编辑 蔡静雯)

(下转第 131 页 continued on page 131)

Actual Range Tree Indexing Scheme for Multi-source Heterogeneous Data Sources

WU Runze¹, CAI Yongtao¹, CHEN Wenwei¹, CHEN Wengang², WANG Yirong³

(1. School of Electrical and Electronic Engineering, North China Electric Power University, Beijing 102206, China;

2. Jincheng Power Supply Company, State Grid Shanxi Electric Power Company, Jincheng 048000, China;

3. Beijing Guodiantong Network Technology Co. Ltd., Beijing 100070, China)

Abstract: Reliable data storage and highly efficient management are needed when big data is generated from power generation, transmission, distribution, and new energy access and consumption. Taking the Energy Internet as a virtual application scenario, this paper proposes a cluster storage scheme for multi-source heterogeneous data. In order to meet the needs of efficient query in the cluster storage, a hierarchical indexing model based on the actual range tree (AR-tree) is developed. And the AR-tree indexing method for multi-source heterogeneous data sources is proposed. This method is based on the double layer index model, in which the global index is developed by taking the actual indexing range of local data as the object. Also, the operation cost of the AR-tree indexing method is comprehensively analyzed. Finally, the simulation results show that the AR-tree indexing method is able to improve the query hit ratio and efficiency.

This work is supported by National Natural Science Foundation of China (No. 51507063).

Key words: energy interconnection; smart grid; big data; index; query; data storage; cluster