

大数据计算的复杂性理论与算法研究进展

李建中^{①*}, 李英姝^②^① 哈尔滨工业大学计算机学院, 哈尔滨 150001^② Department of Computer Science, Georgia State University, Atlanta GA 30302-3965, USA

* 通信作者. E-mail: lijzh@hit.edu.cn

收稿日期: 2016-03-24; 接受日期: 2016-05-31; 网络出版日期: 2016-09-09

国家重点基础研究发展计划 (批准号: 61125106) 资助项目

摘要 最近几年, 大数据研究风起云涌, 取得了许多研究成果. 尽管如此, 大数据基础理论研究结果远未满足实际需要, 大量的关键问题尚未解决, 完整的基础理论尚待建立. 本文旨在探索大数据的基础理论研究问题, 综述大数据计算的复杂性理论和算法的研究进展. 首先, 给出大数据计算的严格定义. 其次, 讨论大数据计算的6个挑战和10个科学技术问题. 然后, 综述大数据计算的复杂性理论和算法设计的研究进展. 最后, 分析评述大数据计算的复杂性理论和算法设计的研究结果, 并给出未来研究方向的思考.

关键词 大数据 大数据计算 大数据计算复杂性 大数据算法

1 引言

信息技术的突飞猛进和人类获取数据技术的迅速发展, 催生了大数据时代的到来. 近年来, 包括我国在内的世界很多国家都在能源、制造业、运输业、信息产业、服务业、科教文化、医疗卫生等领域积累了PB (10^{15} 字节) 级以上规模的大数据. 这些大数据已成为国家战略资源, 引起了政府部门、学术界和工业界的高度重视. 大数据具有如下重大价值:

(1) **带动经济发展.** 国际著名咨询机构 Gartner 在 2012 年发布报告预测, 到 2016 年, 全球大数据相关产业规模将达到 2320 亿美元. McKinsey 全球研究所预测, 仅医疗行业, 大数据将带来每年 3000 亿美元的经济价值. 我国的阿里巴巴和百度等企业通过大数据的分析和挖掘, 已经获得很大经济效益.

(2) **促进社会进步.** 大数据对社会进步具有重要作用. 例如, 利用财政金融等大数据, 可提高政府的管理决策水平和反腐败成效; 利用能源、交通、环境、地理等大数据, 可提高城镇管理水平, 促进能源节省、交通智能化、环境改善等, 实现智慧城市; 利用医疗卫生大数据, 可监测医疗体制运行状况和民众健康变化趋势, 提高全社会医疗水平和民众健康水平, 降低医疗卫生成本; 利用教育大数据, 可以提高全社会教育水平和教育效率.

(3) **加强社会安全和稳定.** 通过对网络大数据的分析挖掘, 能够及时发现社会动态与情绪, 分析舆情, 预警敏感、突发和重大事件, 提高政府的应对能力, 维护社会安全和稳定. 大数据将提供大量就业

引用格式: 李建中, 李英姝. 大数据计算的复杂性理论与算法研究进展. 中国科学: 信息科学, 2016, 46: 1255-1275, doi: 10.1360/N112016-00060

机会, 对社会稳定具有重要作用. Mckinsey 全球研究所预测, 仅美国大数据产业就需 14 万至 19 万工作人员和 150 万数据管理人员.

(4) 变革科学研究模式. 大数据正在加速科学研究模式的变革. 科学研究将从过去的假设驱动型向数据驱动型转化. 通过对大数据的分析挖掘, 科学工作者可以发现新的自然现象和自然规律. 例如, Sloan Digital Sky Survey 数据库已成为天文学研究的核心资源, 天文学家已从该数据库发现大量天文现象和规律. 利用生物大数据进行生物学研究已被广泛接受.

虽然大数据价值重大, 但是仅仅拥有大数据并不能实现其价值. 为了实现大数据的重大价值, 我们需要解决大量的科学技术问题. 大数据获取、大数据传输、大数据存储、基于大数据的问题求解 (包括大数据的查询、挖掘和分析等) 是大数据研究的 4 个重要方面. 我们把这 4 方面的所有计算问题定义为大数据计算问题, 把求解大数据计算问题的过程定义为大数据计算.

最近几年, 大数据计算的研究风起云涌, 取得了很多研究成果. 本文旨在分析大数据计算的挑战和研究问题, 介绍大数据计算的复杂性理论和算法的研究进展, 探索未来的研究方向. 限于篇幅, 本文在介绍研究进展时, 主要集中在 3 个方面, 即大数据计算的复杂性理论, 大数据计算的算法设计方法, 大数据查询、挖掘与分析的算法.

2 大数据计算的挑战和科学技术问题

2.1 大数据计算的挑战

大数据计算问题是指以大数据为输入的计算问题. 大数据计算则是求解大数据计算问题的过程. 大数据的规模 (volume) 大、种类 (variety) 多、速度 (velocity) 快、价值 (value) 大等特点, 向传统计算理论和计算技术提出了很多挑战. 下面介绍 6 个重要挑战.

挑战 1. 传统计算复杂性理论不适于大数据计算. 传统计算复杂性理论把计算问题分为易解 (tractable) 类和难解 (intractable) 类, 难解问题类又分为多个层次. 传统计算复杂性理论假设“易解问题是可由确定的图灵机在多项式时间内求解的问题”, 并在此基础上建立了完整计算复杂性理论. 遗憾的是, 当数据量超过 PB 级时, 多项式时间算法已不能在可容忍时间内求解大数据计算问题. 例如, 使用读写速度为每秒 6 GB (6×10^9 字节) 的固态硬盘驱动器线性扫描 1 PB (10^{15} 字节) 的数据集合, 需要 1.9 天的时间, 而线性扫描 1 EB (10^{18} 字节) 则需要 5.28 年. 显然, 可由确定图灵机在多项式时间内计算的问题是易解问题的假设不再成立, 传统计算复杂性理论的根基被动摇, 传统计算复杂性理论不适于大数据计算.

挑战 2. 现有计算模型不能准确地刻画大数据计算. 计算模型是大数据计算复杂性理论的基础. 虽然目前存在很多计算模型 (如 PRAM 和 Map-Reduce), 但是这些计算模型都假定计算资源无限, 没有考虑大数据计算的资源受限特点, 而且 Map-Reduce 还存在性能低于 PRAM 和编程模式单一的问题. 所以, 现有的计算模型不能准确地刻画大数据计算.

挑战 3. 大量的大数据计算问题的计算复杂性悬而未决. 虽然大数据计算问题求解已经进入人们的眼帘, 但是大数据计算问题的计算复杂性却很少有人问津, 大量的大数据计算问题的计算复杂性悬而未决, 很多疑问困扰着我们: 什么样的问题是易解或难解的? 什么样的难解问题是可近似求解的? 什么样的问题是可并行求解的? 什么样的问题是并行可扩展的? 大数据计算问题如何按照复杂性分类? 这些问题不解决, 大数据计算问题的研究将成为无基础的空中楼阁.

挑战 4. 传统的多项式时间算法不适于求解大数据计算问题. 从上面 1 PB 数据线性扫描的例子

可以看出: 由于大数据的规模巨大, 多项式时间算法已经无力在人们能够容忍的时间内求解大数据计算问题, 从而不适于求解大数据计算问题. 多数现有的数据计算问题的求解算法 (如数据查询、挖掘、分析等问题的求解算法), 都具有多项式时间或更高的计算复杂性, 难以求解大数据计算问题.

挑战 5. 传统计算技术难以满足大数据计算需求. 传统的高性能计算机系统是针对计算密集型计算需求设计的, 数据存放与计算隔离, 数据访问带宽受限, 难以适应大数据计算需要. 这些系统的软件系统、编程模型和软件开发环境也难以满足大数据计算的需求. 传统的数据传输技术也难以适应 TB 级以上大数据的传输要求.

挑战 6. 大数据的低质量严重影响大数据的应用. 随着数据的爆炸性增长, 劣质数据也随之而来, 数据可用性受到严重影响. 数据质量问题必然导致源于数据的知识和决策的严重错误. 由数据质量所导致的知识和决策错误已经在全球范围内造成了恶劣后果, 严重困扰着信息社会, 已经引起了世界各国的不安. 因此, 大数据的质量保障也是一项巨大的挑战.

2.2 大数据计算的研究问题

上述 6 个挑战向我们提出了诸多新的研究问题. 下面是 10 个大数据研究的关键科学技术问题.

问题 1. 研究能够准确刻画大数据计算的计算模型, 满足大数据计算的复杂性理论研究需求. 针对大数据计算的资源受限的特点, 面向现有的各种并行计算系统, 特别是即将出现的新并行计算系统 (如基于非易失存储器的并行计算系统), 建立能够准确刻画大数据计算的新计算模型, 满足大数据计算的复杂性理论研究的需求.

问题 2. 研究大数据计算的复杂性理论, 为大数据查询、挖掘、分析等计算问题的复杂性研究奠定基础. 首先, 建立大数据计算问题易解性判定新标准. 然后, 基于大数据计算问题易解性判定新标准, 针对时间复杂性、空间复杂性、I/O 复杂性和通信复杂性, 建立新的大数据计算复杂性理论, 研究各种大数据计算问题的固有计算复杂性.

问题 3. 研究大数据计算的算法设计方法学. 以亚线性或对数多项式时间为时间复杂性上界, 针对大数据计算问题的精确算法、近似算法、随机算法和并行算法的设计问题, 探索算法设计和分析的新方法.

问题 4. 设计与分析大数据查询、挖掘、分析等问题的求解算法. 大数据查询、分析、挖掘等是大数据应用中使用最广泛的重要操作. 现有算法的时间复杂性皆为多项式级或更高, 不适于大数据. 我们需要在新的易解性判定标准下, 考虑大数据计算的资源受限性, 设计求解问题的有效算法, 特别是具有亚线性或对数多项式时间复杂性的算法, 分析算法的时间复杂性、空间复杂性、通信复杂性和 I/O 复杂性, 并针对时间复杂性、空间复杂性、通信复杂性和 I/O 复杂性的各种组合, 设计最优化算法.

问题 5. 研究大数据获取的理论与方法. 大数据主要来源于互联网 (产生网络大数据)、物联网和传感网 (产生感知大数据)、科学实验 (产生科学大数据) 等多种数据源. 我们需要针对这些数据源及其所产生的各类大数据的特点, 研究数据获取的理论与方法, 特别要解决无价值数据最小化和有价值数据最大化问题.

问题 6. 研究大数据传输的理论与技术. 针对通信网络的规模和异构性迅速增大的状况以及大数据高效跨层跨域传输的需求, 研究: 新的网络体系结构、通信协议、编解码方法; 大数据实时、安全、可靠传输的理论与方法; 大数据传输的调度、控制和评估的理论与方法.

问题 7. 研究大数据存储的理论与方法. 针对网络通信瓶颈问题, 研究存储与计算融合的大数据存储的理论和方法、存储系统的可靠性和安全性、最小化存储系统能耗等理论与方法, 解决数据中心的存储与计算分离所带来的严重传输瓶颈问题.

问题 8. 研究大数据可用性的理论和方法. 针对大数据的质量问题, 研究大数据可用性的表达机理、大数据可用性评估理论和方法、大数据错误自动发现和修复的理论和方法、弱可用 (即低质量) 大数据上的近似计算的理论和算法.

问题 9. 支持大数据计算的计算机系统. 针对现有大规模计算机系统不适于大数据计算的问题, 研究新的可扩展并行计算机系统、实时高效的内存计算技术、高效的存储与计算耦合技术、高效的分布式并行计算技术以及相应的软件系统和软件开发环境.

问题 10. 面向应用的大数据计算的理论与方法. 大数据只有在实际应用中才能体现出大价值. 计算机科学家必须与社会学、生物学、医学卫生等领域的专家合作, 凝练各应用领域的大数据计算问题, 研究这些问题的计算复杂性, 设计求解这些问题的有效算法.

3 大数据计算的计算复杂性理论和算法设计的研究进展

大数据研究可以追溯到 20 世纪 80 年代的大规模科学与统计数据库研究^[1~3]. Conference on Scientific and Statistical Database Management 是当时具有代表性的国际会议, 至今已经持续举办了三十多年. 然而, “大数据”一词则起源于 2008 年 9 月名为 Big Data 的 *Nature* 杂志专刊. 2011 年 2 月 *Science* 杂志也推出专刊 Dealing with Data, 讨论大数据面临的问题和挑战. 至此, 大数据的巨大价值得到了国内外学术界、产业界和政府的高度重视, 大数据研究风起云涌. 尽管如此, 大数据基础研究结果远未满足实际需要, 大量的关键技术问题尚未解决, 完整的基础理论尚待建立, 路漫漫其修远兮, 尚需上下而求索. 下面以大数据计算的计算复杂性理论、大数据计算的算法设计方法、大数据查询挖掘与分析等问题的求解算法为核心, 讨论国内外大数据基础理论研究的现状和发展趋势.

3.1 大数据计算的复杂性理论的研究进展

大数据计算的复杂性理论是大数据计算的基础. 在 2.1 小节的挑战 1 中, 已经看到, 对于大数据计算问题来说, “可由确定图灵机在多项式时间内求解问题是易解问题”的假设不再成立, 传统计算复杂性理论的根基被动摇. 显然, 建立大数据计算的复杂性理论已经成为大数据研究的重要任务. 遗憾的是, 目前的相关研究工作十分初步, 研究结果非常少, 主要包括如下几个方面.

3.1.1 大数据计算问题易解性判定标准

近年来, 李建中针对大数据计算问题的易解性判定标准问题, 开展了一系列实验研究. 通过实验和分析发现, 当数据规模大于 TB 级时, 多项式时间已经不再是人们可容忍的计算时间了, 即具有多项式时间复杂性的大数据计算问题已经不再是易解问题. 下边是 3 组实验结果^[4].

实验 1.

实验环境: HPxw9300 计算机, 含 8 个 64 位 CPU, 32 G 内存, 2.4 TB 硬盘.

计算问题: 输入: 两个 1 TB 数据集合 T 和 S; 输出: T 的选择、聚集、投影, T 和 S 连接的结果.

计算时间: 选择的计算的时间是 9.9 小时, 聚集的计算时间是 9.98 小时, 投影的计算时间是 10.2 小时, T 和 S 连接的计算时间是 49.8 小时.

实验 2.

实验环境: 与实验 1 相同.

求解问题: 输入: 具有 16 个维属性的 1TB 多维数据集 T; 输出: 使用最快的 Fastcube 算法在 T 上计算 Cube 的结果.

计算时间: 89.89 小时.

实验 3.

实验环境: 具有 64 个计算节点的 Cluster, 每个节点包括 2 个 4 核 CPU、32 G 内存和 1 TB 硬盘, 计算节点间通信带宽峰值为 1 G/s.

求解问题: 输入: 两个 10 TB 数据集 T 和 S; 输出: T 和 S 的 Join 结果.

计算时间: 68.484 小时.

这些实验说明, 传统计算问题易解判定标准“多项式时间”不再适用于大数据计算. 于是, 李建中猜测亚线性时间 ($O(n^{1/k})$) 或对数多项式时间 ($O(\log^k n)$) 可能是大数据计算问题易解性的判定标准^[5~8], 得到了国内外同行的认可, 其中 n 是问题输入大小, k 是常数. 但是, 这个结论尚需更多的实验研究来验证.

3.1.2 大数据计算模型

计算模型是研究可计算性理论和计算复杂性理论的基础. 计算模型已经具有 80 年的研究历史. 20 世纪 30 和 40 年代, 针对问题的可计算性, Turing 提出了图灵机计算模型^[4], Kleene 提出了递归函数计算模型^[9], Post 提出了 Post 系统^[10], Church 提出了 λ -演算计算模型^[11], Church 和 Turing 证明了这 4 种计算模型的等价性, 提出了著名的 Church-Turing 命题^[12~15]. 由于图灵机模型更接近于现代计算机系统, 至今仍被广泛使用. 随着现代计算机系统的出现, 人们又提出了更接近于实际计算机系统的计算模型, 如随机存取机器模型 RAM^[16,17]. 针对并行计算机系统, 很多并行计算模型在 20 世纪 60 和 70 年代被提出, 其中布尔网络^[18,19]和 PRAM^[20]是具有代表性的并行计算模型. 在这两个模型的基础上, 人们还提出了一些更实际的并行计算模型, 如 APRAM、BSP、LogP、C₃ 等并行计算模型^[21].

上述计算模型都有一个共同的特点, 即假定计算资源无限. 所以, 上述计算模型都不能刻画大数据计算的计算资源受限的特点. 同时, 这些计算模型也不支持大数据计算问题的 I/O 复杂性和通信复杂性的分析.

目前已经出现了很多大数据计算系统, 如 Hadoop 系统、Spark 系统、Storm 系统等. 但是, 目前针对大数据计算的计算模型的研究还很少. 2010 年, Karloff 等对 Hadoop 和 Spark 大数据计算系统采用的 Map-Reduce 计算模式进行了精致的抽象, 提出了 Map-Reduce 计算模型^[22]. Map-Reduce 计算模型是目前唯一一个针对大数据计算的计算模型. Karloff 等在文献^[22]中比较了 Map-Reduce 计算模型和 PRAM 模型的计算能力, 证明了: 任意一个使用 $O(n^{2-2\epsilon})$ 空间和 $O(n^{2-2\epsilon})$ 个处理器的 $T(n)$ 时间 CREW PRAM 程序都可以由一个迭代次数为 $O(T(n))$ 的 Map-Reduce 程序模拟, 其中 n 是输入数据集的大小. 从这个结果可知: 如果 Map-Reduce 计算的每次迭代至少需要 $O(f(n))$ 时间, Map-Reduce 执行一个 $T(n)$ 时间 PRAM 程序的时间为 $O(f(n)T(n))$. 于是, Map-Reduce 的计算效率低于 PRAM. 如果 $f(n)$ 是高于亚线性或对数多项式的函数, 则 Map-Reduce 难以实现亚线性或对数多项式时间计算. 同时, Map-Reduce 仅支持迭代编程模式, 很难有效支持不适于迭代计算的大数据计算问题的求解. Map-Reduce 既不能刻画大数据计算的资源受限的特点, 也不支持大数据计算的 I/O 复杂性和通信复杂性分析.

3.1.3 基于 Map-Reduce 模型的计算复杂性理论

自从 Map-Reduce 计算模型^[22] 建立以来, 出现了一些基于该模型的计算复杂性理论研究工作, 包括 Map-Reduce 模型与其他计算模型的关系、Map-Reduce 模型的通信复杂性、基于 Map-Reduce 的计算问题的复杂性.

Map-Reduce 模型与其他计算模型的关系. Karloff 等比较了 Map-Reduce 模型与 PRAM 模型的计算能力, 证明了很多 PRAM 可解问题都是 Map-Reduce 可解的^[22]. Karloff 等在文献 [22] 中定义了问题类 $DMRC = \{p \mid p \text{ 由迭代次数为 } O(\log^k n), \text{ 每次迭代时间为 } O(\log^k n) \text{ 的 Map-Reduce 程序求解}\}$, 并证明了 $DMRC \subseteq P$ 以及 $DMRC \subseteq NC$ 如果 $P \subseteq NC$.

Map-Reduce 模型的通信复杂性研究. Afrati 等研究了 Map-Reduce 模型的通信复杂性问题^[23], 提出了确定大数据计算问题在 Map-Reduce 模型下通信复杂性下界的模型, 给出了解决并行性和通信复杂性之间的均衡问题的方法. Tao 等提出了同时最小化 Map-Reduce 的内存空间、CPU 时间和通信时间的 Minimal 算法的概念, 证明了很多数据库基础问题存在 Minimal 求解算法^[24].

基于 Map-Reduce 模型的大数据计算问题的复杂性. 文献 [25~27] 分析了 Join 查询、Datalog 查询和传递闭包在 Map-Reduce 模型下的计算复杂性, 特别是通信复杂性. 文献 [28] 分析了子图查询在 Map-Reduce 模型下的计算复杂性. 文献 [29] 分析了 Fuzzy Joins 问题在 Map-Reduce 模型下的计算复杂性. 文献 [30] 分析了 Skyline 查询在 Map-Reduce 模型下的计算复杂性. 文献 [31] 分析了关系查询在 Map-Reduce 模型下的通信复杂性. 文献 [23] 确定了基于海明距离的近邻查询、图数据三角形模式查询、大矩阵乘法这 3 个大数据计算问题在 Map-Reduce 模型下的通信复杂性下界和上界.

虽然人们开展了基于 Map-Reduce 模型的大数据计算问题的计算复杂性研究, 但是研究工作刚刚起步, 还有大量的大数据计算问题的复杂性问题尚未解决.

3.1.4 超越 Map-Reduce 模型的大数据查询的计算复杂性

Fan 等在文献 [32] 中研究了大数据查询的计算复杂性问题. 首先, 定义了基于对数多项式时间的大数据查询 DB- 易解性, 提出了 DB- 易解的大数据查询类 $\Pi T_0 = \{Q \mid \text{查询 } Q \text{ 的预处理时间为多项式, 处理时间为对数多项式时间}\}$, 并证明了关系数据库上的 Selection 查询、无序表上的元素查询等多种查询都是 DB- 易解的. 然后, 定义了对数多项式时间规约, 记作 NC- 规约, 并在此基础上定义了另一个大数据查询类 $\Pi T = \{Q \mid \text{查询 } Q \text{ 能用 NC- 规约转化为 DB- 易解的查询}\}$. 其次, 基于 NC- 规约定义了大数据查询的 ΠT_0 - 完全性和 ΠT - 完全性, 并证明了 ΠT_0 - 完全问题和 ΠT - 完全问题的存在性. 最后, 证明了 (1) $P \subseteq \Pi T$; (2) PTIME 查询类 $\subseteq \Pi T$; (3) 无 NP- 完全问题属于 ΠT 除非 $P=NP$; (4) 存在查询 $Q \in P, Q \notin \Pi T$ 除非 $P=NC$.

Fan 等的工作和前面李建中的工作是目前超越 Map-Reduce 的大数据计算复杂性理论方面的两项工作. 但是, 这些工作仅仅是对大数据计算问题易解性和大数据查询复杂性的一点探讨, 相对于完整的大数据计算的复杂性理论来说, 仅仅是大海里的一束浪花. 大数据计算的复杂性理论研究任重道远.

3.1.5 小结

(1) 实验表明, 多项式时间不适于作为大数据计算问题的易解性判定标准. 但是, 亚线性时间或对数多项式时间是易解性判定标准的合理性尚需更多更广的实验来验证.

(2) 传统计算模型不能准确描述大数据计算, 特别是大数据计算的资源受限特点, 不能有效地支持大数据计算的 I/O 复杂性和通信复杂性的分析.

(3) Map-Reduce 计算模型具有计算效率低和编程模式单一的问题. 目前仅有少数问题的 Map-Reduce 计算复杂性得到了解决, 基于 Map-Reduce 模型的复杂性理论的诸多方面还没有被考虑.

(4) 超越 Map-Reduce 的大数据计算复杂性理论方面的工作刚刚开始, 仅对大数据计算问题易解性判定标准和大数据查询问题进行了初步探讨. 目前尚未见到针对 Storm、Cluster、基于非易失存储器的并行计算机等大数据计算系统的计算模型和计算复杂性理论方面的研究. 大数据计算的复杂性理论研究任重道远.

3.2 大数据计算问题求解算法的设计方法

多年来, 人们在求解大规模数据计算问题的过程中, 总结出了如下一些大数据计算的算法设计方法.

3.2.1 大数据的 Map-Reduce 计算方法

Dean 等针对大规模 Cluster^[33], 提出了支持大数据计算的 Map-Reduce 计算模式, 并在 Hadoop 和 Spark 等系统中得以实现¹⁾.

这种方法已经成为大数据计算的重要模式, 得以广泛应用^[24, 28, 29, 34~48]. 然而, 这种计算模式具有效率低和编程模式单一的缺点.

3.2.2 大数据的压缩计算方法

大数据计算问题的难解性源于数据规模庞大. 传统的计算方法单纯追求降低算法的时间复杂性, 没有考虑缩减数据规模, 对复杂性下界较高的大数据计算问题的计算效率提高甚微. 如果把大规模数据计算问题转换为小数据计算问题, 则计算效率会显著提高. 为此, 李建中等突破传统思维, 通过缩减数据规模来降低计算开销, 提出大规模数据的压缩计算的方法: 首先以预处理的方式压缩数据集合, 然后直接在压缩数据上无解压地完成计算.

李建中等首先针对传统数据压缩方法不能支持压缩数据上无解压计算的问题, 研究了支持无解压计算的新数据压缩原理, 发现并验证了面向关系数据的压缩方法能够支持无解压计算的充分条件“映射完全性”, 还发现了映射的时间复杂性对无解压计算的效率具有重要影响, 并以降低映射时间复杂性为目标, 提出了一系列支持无解压计算的数据压缩方法^[49~51].

然后, 基于上述数据压缩方法, 李建中等针对转置、聚集、Cube 等大规模数据分析问题, 提出了一系数据压缩计算的算法^[49, 52~54], 揭示了大规模数据压缩计算的原理.

近年来, Fan 和李建中等还开展了大规模图数据的压缩计算研究, 取得了一些研究成果^[55, 56]. 这些算法的理论分析和实验结果表明, 压缩计算方法极大地提高了大规模数据计算的效率.

当大数据不可压缩或压缩比较低时, 大数据的压缩计算方法将失效. 该方法尚需扩大应用范围, 在更多的大数据计算问题求解过程中验证和改善.

3.2.3 大数据的 (ϵ, δ) - 近似计算方法

(ϵ, δ) - 近似计算方法也是把大规模数据计算转换为小数据计算的计算方法. 李建中和程思瑶等以大规模数据聚集问题为切入点, 提出了 (ϵ, δ) - 近似计算方法: 首先在大数据上随机抽取一个小数据集, 然后在小数据集上求解计算问题, 使得计算误差大于 ϵ 的概率小于 δ . 他们的论文^[57~59]揭示

1) Hadoop. <http://hadoop.apache.org/>.

了 (ϵ, δ) - 近似计算方法的原理, 解决了这种方法的 4 个关键问题: 抽样方法选择问题, 样本最小化问题, 对于给定问题的 (ϵ, δ) - 数学求解器的构造问题, 样本动态维护问题.

由于采用了随机抽样方法, 很多个体数据被抽取到的概率很低, 从而 (ϵ, δ) - 近似计算方法仅适用于宏观分析, 不适于个体数据的微观查询. 这种方法的应用范围目前还不广泛, 尚需要通过大量的大数据计算问题的求解来验证和改善.

3.2.4 基于支配数据集的大数据计算方法

大数据虽然量大, 但是与每个计算问题相关的数据量却不一定很大. 大数据中与一个计算问题相关的数据集定义为求解这个问题的支配数据集. 李建中和程思瑶等提出了基于支配数据集的大数据计算方法, 给出了实现大规模数据计算问题转换为小数据计算问题的新方法^[60]. 给定一个大数据计算问题, 基于支配数据集的计算方法如下: 首先在大数据集合中抽取支配数据集, 然后在支配数据集上完成计算. 这里的关键问题是支配数据集的抽取. 文献 [60] 给出了目前唯一的一个支配数据集抽取方法. 基于支配数据集的计算方法是一个有前途的方法, 但是目前的研究才刚刚开始, 大量工作需要深入开展.

3.2.5 大数据的增量式计算方法

增量式计算方法是一个传统的方法. 这种方法可以用于大数据计算, 把大数据计算转化为一系列的小数据计算问题. 它可通过把大数据集合划分为任意多个子集合, 递增地求解出大数据计算问题越来越精确的解. 李建中等较早地应用增量式计算方法求解数据流挖掘聚集问题和大数据聚集查询问题^[61,62]. 最近几年, Fan 等应用增量式计算方法求解大图匹配问题和大规模数据错误发现问题^[63,64], 取得了很好效果. 然而, 这种方法要求欲计算问题必须满足可加性, 限制了它的应用范围.

3.2.6 I/O 高效计算方法

目前的大数据计算平台多数都建立在“内存 – 磁盘”两层存储结构的基础之上. 在这种系统结构下, I/O 是数据密集型计算的严重瓶颈. I/O 高效算法设计方法的目标是: 设计可以在“内存 – 磁盘”两层存储结构上高效运行的算法. 长期以来, I/O 高效算法一直是数据库领域关心的问题, 所有的数据库操作的算法设计都以 I/O 高效为主要目标. 当我们面对大数据的时候, I/O 瓶颈问题更加突出, I/O 高效算法的设计更加重要. 文献 [65] 以 B- 树索引为例, 提出了一种 I/O 高效索引的方法. 文献 [66] 以最短路径为例, 提出了一种图数据上 I/O 高效算法的设计方法. 文献 [67] 提出了一种求解平面点集合的 Range Skyline 问题的 I/O 高效算法设计方法. 值得注意的是, 目前的大数据计算研究对算法的 I/O 效率关心不多.

3.2.7 小结

(1) 大数据计算的算法设计方法非常重要, 目前的研究结果还很少. 大数据计算的算法设计方法难以通过抽象的方法得到, 必须在求解具体的大数据计算问题的过程中提炼而得.

(2) 现有的大数据计算的算法设计方法各有优点和局限, 试用的范围还很狭窄, 尚需在更广泛的大数据计算问题求解过程中验证和改善.

(3) 现有的大数据计算的算法设计方法很少考虑通信复杂性和 I/O 复杂性, 更没有考虑如何设计具有亚线性或对数多项式时间复杂性的算法.

3.3 大数据查询处理算法

大数据查询是观察了解大数据所描述的物理世界的基本方法, 大数据查询处理算法则是大数据计算的最基本算法和大数据应用的重要基础, 引起了人们的密切关注, 取得了如下一些研究成果.

3.3.1 基于 Map-Reduce 的大数据查询处理算法

文献 [29, 35~38] 研究了在大数据背景下, 如何利用 Map-Reduce 处理经典的关系查询. 关系查询的基础是关系代数操作. 目前关系查询处理研究主要集中在连接操作的算法设计方面, 其他操作的研究不多. 文献 [35] 应用 Map-Reduce 方法处理连接操作, 提出了基于连接条件分解的计算方法. 文献 [38] 进一步考虑了多路 Theta- 连接操作的实现算法, 提出基于任务调度的处理算法. 文献 [29] 在单轮 Map-Reduce 计算模式下考虑模糊连接操作的计算问题, 提出了 Mapper、Reducer 和通信 3 类代价模型, 通过分析比较现有的模糊连接算法, 发现没有一个算法能在通信代价和 Reducer 代价上同时优于其他所有算法. 文献 [36, 37] 分别提出了处理 Top-k 和 k- 近邻查询的 Map-Reduce 算法. 文献 [24] 定义了 Map-Reduce 环境下的最优化算法, 即具有单机空间代价 $O(n/t)$ 、单轮迭代单机通信代价 $O(n/t)$ 、常数迭代次数、单机时间代价 $O(T_{seq}/t)$ 的算法, 其中 n 是问题的大小, t 是机器数目, T_{seq} 是单机解决大小为 n 的问题的时间代价, 并提出求解排序和聚集等问题的接近最优化的算法. 文献 [68] 研究了分布式环境下树结构数据的查询处理问题, 算法迭代次数与数据大小成线性关系. 文献 [69] 研究了大数据上的相似连接问题, 提出了 Map-Reduce 环境下的高效的算法.

索引是数据库管理系统中加速查询处理的重要技术. 文献 [34, 39] 将索引技术移植到 Map-Reduce 平台, 提出了云计算系统中的数据索引技术.

3.3.2 基于 Cluster 的非 Map-Reduce 大数据查询处理算法

文献 [70] 研究了基于 Cluster 的基数排序问题. 文献 [71] 研究了基于 Cluster 的大数据连接操作的分布式计算方法, 提出了时间复杂性达到了问题固有复杂性下界的优化算法. 文献 [72] 研究了数组结构大数据的抗偏斜连接查询处理问题, 提出了连接查询处理的分布式优化计算框架: 首先选择具体的查询算法, 然后利用代价模型选择计算节点并分配计算任务. 文献 [73] 研究了分布式计算环境下的连接查询的优化处理问题, 优化目标是通信代价. 文献 [74] 研究了并行计算环境下复杂分析查询的优化问题, 主要思想是对复杂查询中的公共查询子表达式进行优化. 与上述工作不同, 文献 [75, 76] 关注查询问题本身的难度, 研究集中式环境下如何利用剪枝策略提高大数据查询的处理效率, 提出了基于剪枝策略的查询处理算法. 文献 [77] 研究了非等值连接操作的问题, 利用编码技术以及 Bloom Filter 等过滤方法快速获得查询结果.

3.3.3 资源受限的大数据查询处理方法和并行可扩展算法

现有计算资源相对于大数据查询来说十分有限, 使大数据查询处理陷入困境. 如何在资源受限的条件下处理大数据查询是十分困难的问题. Fan 等研究了资源受限的大数据计算方法^[78~81]. 文献 [79] 提出了用小数据回答大数据查询的方法. 文献 [80] 提出了 Scale-Independent 的概念: 查询 Q 在数据 D 上相对 M 是 Scale-Independent 的, 当且仅当 $Q(D)$ 可以基于 D 中至多为 M 个数据的子集合求解, 并确定了 Scale-Independence 判定问题的计算复杂性. 文献 [78, 81] 研究了资源受限的大图数据模式查询和可达性查询的处理问题, 识别出一类可以由小数据求解的大数据查询, 提出了给定查询是否可由小数据求解的判定算法, 并研究了资源受限查询问题的计算复杂性, 提出了精确和近似的资源受

限查询处理算法. 文献 [78] 还基于动态数据缩减的思想, 提出了资源受限的近似计算算法, 实验表明: 把 PB 级数据缩减到 GB 级, 算法仍然能够保持很高的精度. 文献 [78] 研究了资源受限的大图查询处理问题, 确定了图模式匹配查询和可达查询问题的计算复杂性, 提出了精确和近似的求解算法.

算法的并行可扩展性 (即算法的时间计算复杂性随计算节点数的增加而单调递减) 一直是并行计算的重要问题. Fan 等研究了大数据计算问题的可扩展性 [82, 83]. 文献 [82] 提出了大数据计算的并行可扩展性概念, 证明了不存在求解图 Simulation 问题的并行可扩展算法. 文献 [83] 提出了一个并行可扩展的图相关规则查询算法.

3.3.4 大图数据的查询处理算法

文献 [84, 85] 研究了云计算系统中在 RDF 图数据上处理 SPARQL 查询的问题, 提出了高效的处理算法. 文献 [86, 87] 提出了利用索引等技术, 提高图数据查询效率的算法. 文献 [88] 定义了图上的 SRJ 查询, 并提出了基于数据分解的查询处理算法. 文献 [89, 90] 针对图数据访问的特点, 研究了大图数据存储问题, 提出了高效的图数据管理系统. 文献 [91] 提出了流模型上的概要数据获取方法, 通过缩减数据来支持图查询的优化. 文献 [92] 研究了大图数据上的子图枚举问题, 利用基于迭代计算的方法避免了直接处理连接操作. 文献 [93] 研究了线图数据集上的图匹配搜索问题, 利用保持距离和结构不变的映射, 把图分组, 降低了图匹配操作的数目. 文献 [94] 研究了分布式动态图数据上的连续最短路径查询问题, 解决了连续最短路径查询的中间结果维护问题. 文献 [95] 研究了大图数据上的最短路径问题, 主要思想是利用预计算技术, 获得一个空间代价较小的数据结构, 然后利用这个小数据结构快速地近似求解最短路径查询. 文献 [96] 研究了基于 Map-Reduce 的大图查询问题, 提出了迭代次数为 $O(\log n)$ 、每次迭代通信代价为 $O(n + m)$ 的连通分量查询和最小生成森林查询的处理算法. 文献 [97~99] 分别研究了基于抽样的大图数据总量查询处理问题、动态大图数据上的可达性查询处理问题、应用关系数据管理技术解决大图数据的管理等问题. 文献 [100] 研究了在不用索引的情况下, 如何利用并行计算处理具有 10 亿顶点的大图数据匹配问题. 文献 [101] 针对现有可达性查询索引无法处理大数据的问题, 提出了新的可达性查询框架, 改进了可达性查询索引, 提高了可达性查询的处理速度. 文献 [102~104] 分别提出了求解大图数据链路预测问题的有效算法、求解最大团问题的可扩展算法、基于外存求解所有强连通分量的算法. 文献 [105] 研究了大图数据上的三角形计数和枚举问题, 在多核计算环境下设计了高效的精确和近似算法, 时间复杂性为 $O(|V| \log |V| + |E|)$, 其中 V 是图的节点集合, E 是边集合. 文献 [106] 提出了分布式环境下的连通分量计算方法, 解决了 Map-Reduce 框架下迭代次数与迭代间通信代价的权衡问题, 其迭代次数为 $O(\log n)$, 每次迭代的通信代价 $O(|V| + |E|)$, n 是最大连通分量的大小. 文献 [28] 在 Map-Reduce 计算环境下, 利用将模式图转化为多路连接操作的思想, 提出了子图枚举的高效算法. 文献 [103] 在 Map-Reduce 计算环境下, 利用数据划分策略和分支界限方法, 提出了最大团求解算法. 文献 [78~83] 提出了大图数据查询的资源受限处理算法和并行可扩展算法, 详见 3.3.3 小节.

3.3.5 小结

(1) 虽然大数据查询处理的结果相对较多, 但是研究问题的范围较窄, 关系大数据上的查询主要集中在连接操作的算法设计方面, 大图数据查询主要集中在子图模式查询、最短路径、最大连通分量、最大团等基本图操作的算法设计方面, 研究的范围尚需扩大.

(2) 大数据查询处理研究主要围绕 Cluster 和 Map-Reduce 计算平台和系统开展, 没有考虑正在出

现的新并行计算系统, 如基于非易失存储器的并行计算系统, 研究结果受到 Cluster 和 Map-Reduce 局限性的限制.

(3) 大数据查询问题的固有复杂性分析还很少见, 求解算法的分析主要考虑了时间复杂性, 没有考虑 I/O 复杂性, 很少考虑通信复杂性.

(4) 多数大数据查询处理算法的时间复杂性是多项式时间或更高, 亚线性或对数多项式时间算法还很少见.

3.4 大数据挖掘算法

大数据挖掘是认知大数据所描述的物理世界和发现物理世界内在规律的主要方法. 大数据挖掘算法是大数据计算和大数据应用的关键之一. 虽然数据挖掘的研究已经具有很长的历史, 但是大数据挖掘算法的研究工作还不多, 研究成果很少. 下面是大数据挖掘算法的主要研究成果.

3.4.1 基于 Map-Reduce 的大数据挖掘算法

文献 [40] 基于 Map-Reduce, 提出了频繁序列挖掘算法, 利用 “gap constraints” 把算法的输出限制在一个较小子集上, 解决了数十亿条记录的频繁模式挖掘问题. 文献 [107] 基于同样思想, 在数据存在层次关系的情况下, 提出了求解频繁模式挖掘问题的有效算法. 文献 [41] 基于 Map-Reduce, 设计了 SystemML 系统, 允许用户以高级语言的方式描述机器学习算法, 并支持自动地将用户编写的脚本程序转换为 Map-Reduce 计算任务流, 从而实现复杂的机器学习算法. 文献 [108] 研究了大数据上的频繁模式挖掘问题, 利用 Map-Reduce, 在整个搜索空间的不同部分进行并行搜索, 提升了挖掘算法的性能. 文献 [42] 研究了聚簇分析问题, 提出了流数据聚簇分析方法, 该方法能够根据任务需要, 动态地将数据分配到各个处理单元, 平衡了各处理单元的工作负载.

3.4.2 基于随机抽样的大数据挖掘算法

文献 [109] 研究了大数据采样问题, 提出了一个基于数据划分的高效采样算法, 支持基于抽样的大数据挖掘. 文献 [110] 在 Map-Reduce 计算环境下, 提出了基于随机抽样的大数据频繁项集挖掘的近似算法. 文献 [111] 使用渐进采样思想, 提出了基于随机抽样的大数据频繁项集挖掘的近似算法. 文献 [112] 在图的拓扑结构未知情况下, 对边进行随机抽样, 实现了社交网络中主题信息的挖掘. 文献 [113] 提出了基于随机抽样的主成分分析算法, 把低维空间表示和抽样技术相结合, 减少了算法的通信量. 文献 [114] 利用最优抽样技术, 提出累加随机坐标下降算法, 加快收敛速度, 解决复合型极值这个数据挖掘和机器学习中的共性问题, 支持数据挖掘和机器学习. 文献 [115] 提出了一种分布式 SGLD (stochastic gradient Langevin dynamics) 方法, 采用分治的抽样方法, 设计了基于抽样的大文本数据的主题发现算法.

3.4.3 大图数据的挖掘算法

大图数据挖掘算法的研究主要针对不同的应用背景开展, 工作比较分散. 文献 [116] 基于 Hadoop 平台, 利用推理方式, 设计了 HA-LFP 系统, 支持从十亿节点级的大图数据中挖掘有用信息. 文献 [117] 使用 Map-Reduce, 设计了两种算法, 解决社交网络中的匹配问题. 文献 [118] 基于 Map-Reduce, 设计了求解个性 PageRank 问题的 Monte Carlo 近似算法. 文献 [37] 设计了 GraphLab 系统, 支持异步、动态、数据并行等计算方式, 提供了支持机器学习算法的图数据处理框架. 文献 [119] 研究了大数据上的

Dominant 聚簇问题, 提出了时间代价为 $O(mn)$ 和空间代价为 $O(m^2)$ 的算法, 其中 m 是最大簇的大小, n 是图的大小. 文献 [120] 提出了 Map-Reduce 环境下的频繁子图挖掘算法. 文献 [83] 提出了并行可扩展的大图相关规则挖掘算法.

3.4.4 小结

- (1) 大数据挖掘的研究工作不多, 比较分散, 大量挖掘问题尚待解决.
- (2) 大数据挖掘的研究主要围绕 Map-Reduce 开展, 没有考虑其他计算系统和正在出现的新并行计算系统.
- (3) 大数据挖掘问题的固有复杂性分析还很少见, 求解算法的分析仅考虑了时间复杂性, 没有考虑 I/O 复杂性和通信复杂性.
- (4) 多数大数据挖掘算法的时间复杂性是多项式时间或更高, 没有考虑亚线性或对数多项式时间算法的设计与分析问题.

3.5 大数据分析算法

数据分析不同于数据挖掘. 数据分析首先假设该数据可能满足的规律, 然后验证这个规律是否成立, 如果成立则辨识出规律的参数. 数据挖掘则是在没有任何假设的前提下, 发现未曾发现的规律. 大数据分析也是认知大数据所描述的物理世界内在规律的主要方法. 人们在大数据分析算法方面取得了一些研究成果. 下面介绍大数据分析算法的国内外研究进展.

3.5.1 基于 Map-Reduce 的大数据分析算法

文献 [44] 基于 Map-Reduce, 提出了从大数据中抽取满足给定谓词条件的样本数据的算法. 文献 [45] 提出了基于 Map-Reduce 的大数据的小波采样算法. 文献 [47] 借鉴物化和内存替换策略等经典优化方法, 解决了无法装入内存的 Factor 图上的 Gibbs 采样问题. 文献 [28] 提出了从大图数据中抽取匹配给定样本图的所有图的算法. 文献 [48] 研究了核密度估计 (kernel density estimate) 这个重要的数据分析基础问题, 提出了随机和确定两类求解算法, 性能优于已有算法多个数量级. 文献 [43] 基于 Map-Reduce, 提出了大数据在线聚集算法. 文献 [121] 基于 Map-Reduce, 研究了流数据的集合关系分析问题, 提出了基于数据划分、冗余存储和计算负载平衡的高性能并行算法. 文献 [46] 基于 Map-Reduce, 研究了大型矩阵分析问题, 建立了基于 Hadoop/HDFS 的矩阵分析系统. 文献 [122] 基于 Map-Reduce 建立了 OLAP 分析系统.

3.5.2 大数据分析的压缩计算方法

大数据分析的压缩计算方法是指: 首先对大数据进行压缩预处理, 然后直接在压缩数据上进行无需解压缩的分析计算. 文献 [45] 提出了一种支持大数据压缩计算的高效数据压缩方法. 文献 [123] 提出支持联机分析的压缩计算的数据压缩方法. 文献 [52] 提出了 4 种求解多维数据转置问题的压缩计算方法. 文献 [53] 在数据仓库上给出了求解聚集分析问题的压缩计算算法. 文献 [49, 124] 提出了压缩数据上数据分析查询处理的优化技术. 文献 [125] 基于 Cluster, 研究了统计聚集函数分布式计算问题, 提出了基于压缩感知的求解算法, 把通信代价降低到 $O(sk \log N)$, 其中 N 是数据的大小, s 是与数据稀疏程度相关的变量, k 是常数.

3.5.3 基于抽样的大数据分析算法

文献 [57] 基于随机抽样技术, 提出了聚集分析的 (ε, δ) - 近似计算方法. (ε, δ) - 近似计算方法是指在大数据中随机抽取计算所需的样本数据, 使得在该样本上计算结果的误差超过 ε 的概率小于 δ . 这个方法的本质是通过小数据计算来任意近似大数据上的精确计算结果. 文献 [57] 提出了基于均匀抽样的聚集操作的 (ε, δ) - 近似计算方法, 其时间复杂性为 $O(\varepsilon^{-2} \ln(\delta^{-1}))$. 文献 [58, 59] 提出了 4 种基于 Bernoulli 抽样的聚集操作的 (ε, δ) - 近似计算方法, 其时间复杂性为 $O(\ln(1/\delta)1/\varepsilon^2)$.

3.5.4 大图数据的分析算法

文献 [126] 研究了大图数据上的 k - 边连通分量的计算问题, 利用图分解的思想提出了一个图计算框架, 将已有算法的时间复杂性 $O(|V|^2|E| + |V|^3 \log |V|)$ 改进为 $O(hl|E|)$, 其中 E 是图的边集合, V 是图的顶点集合, h 是算法使用的分解树的高度, l 远小于 $|V|$. 文献 [127] 提出了图数据流上的事件匹配算法, 可以在常数时间内处理边的更新. 文献 [128] 研究了大图聚簇问题, 提出了时间代价为 $O((2-c)|E|/(2a-c))$ 和空间代价为 $O(|E| + |V| + l)$ 的算法, 其中 $a = |V|/|E|$, c 是与簇结构相关的系数, l 是算法在局部监测阶段发现的所有局部聚簇大小的和. 文献 [129] 研究了大图的最大独立集问题, 在内存能够容纳所有顶点但容纳不下所有边的条件下, 提出了解决该问题的贪心算法. 文献 [130] 提出了冰山立方体的概念, 利用随机游走而非传统聚集函数来实现大图分析. 文献 [131] 把已有的冰山立方体扩展到大图数据上, 提出新计算框架并且设计了基于随机游走的算法. 文献 [132] 研究了大图数据上的社区发现问题, 利用局部搜索而不是全局搜索的方法求解该问题. 文献 [133] 提出了动态大图数据上 k -truss 社区发现问题的求解算法.

3.5.5 大矩阵计算方法

矩阵计算是数据分析的基础, 大矩阵计算是与大数据分析密切相关的重要研究问题. 文献 [134] 在分布式计算环境下, 研究了矩阵主成分分析算法, 分析了基于不同原理的主成分分析算法的时间代价和通信代价. 设问题的输入是 $N \times D$ 矩阵, 输出是具有 d 行的主成分, 则基于特征分解的求解算法的时间代价为 $O(ND \times \min(N, D))$, 通信代价为 $O(D^2)$; 基于奇异值分解的求解算法的时间代价为 $O(ND^2 + D^3)$, 通信代价为 $O(\max((N+D)d, D^2))$; 基于随机奇异值分解的求解算法的时间代价为 $O(NDd)$, 通信代价为 $O(\max(Nd, d^2))$; 基于概率奇异值分解的求解算法的时间代价为 $O(NDd)$, 通信代价为 $O(Dd)$. 作者还在 Hadoop 和 Spark 系统中实现了部分算法. 文献 [135] 研究了分布式计算环境下求解矩阵运算的优化问题, 通过分析矩阵之间的依赖关系, 设计了分布式环境下矩阵运算的代价估计模型, 提出了为基本操作选择通信代价优化的计算计划的算法、为矩阵计算选择计算计划的算法以及优化计划的分布式下发算法. 文献 [46] 研究了基于矩阵计算的大数据分析的平台搭建问题, 设计了基于 Hadoop/HDFS 计算平台. 文献 [136] 研究了分布式矩阵计算问题, 通过维护一个非常小的矩阵, 有效计算整个矩阵的范数, 通信代价为 $O((m/\varepsilon) \log(bN))$, 其中 N 是数据流的大小, ε 是给定的误差界限, b 是矩阵任意行的范数的上界.

3.5.6 流数据分析

文献 [137] 研究了分布式数据流上 Skyline 分析的动态求解问题, 提出了高效的 Skyline 结果动态维护算法, 并给出了一系列的优化技术. 文献 [138] 针对常用的基于距离定义的离群点检测问题, 提出

了一种通用的计算框架. 文献 [139] 研究了感知大数据上的异常检测问题, 利用压缩的数据结构存储数据流上的历史信息, 快速判定异常. 文献 [140] 研究了并行环境下流数据上的事件匹配问题.

3.5.7 其他大数据分析算法

文献 [141] 研究了使用大规模 Cluster 处理复杂分析查询的问题, 设计并实现了一个类似于 Map-Reduce 的计算框架, 并提供了容错的复杂分析处理能力. 文献 [142] 研究了社交大数据中的观点分析问题, 提出了一个多粒度的观点分析系统, 利用在线的增量式方法, 维护求解该问题的关键数据结构. 文献 [126] 提出了大数据上的 Gibbs 采样算法. 文献 [47] 研究了大数据上的核密度估计问题, 提出了 $O(n \log n)$ 的算法, 并给出了集中式和 Map-Reduce 实现方法. 文献 [93] 研究了大数据上 SimRank 计算问题, 提出了在线和离线两种分布式 Monte Carlo 算法, 离线和在线算法分别具有线性时间和常数时间代价. 文献 [143] 研究了社交大数据上的信念传播问题, 提出了一种线性信念传播算法. 文献 [144] 研究了 Skyline 和 Reverse Skyline 操作在 Map-Reduce 环境下的求解问题, 提出了高效的并行算法.

3.5.8 小结

(1) 大数据分析的研究工作主要集中在抽样、聚集、OLAP 分析、矩阵计算等基本分析操作上, 很多复杂分析操作还未涉及, 大量问题尚待解决.

(2) 大数据分析问题的固有复杂性分析还很少见, 算法分析主要考虑了计算复杂性, 没有考虑 I/O 复杂性, 很少考虑通信复杂性.

(3) 多数大数据分析算法的时间复杂性是多项式时间或更高, 没有考虑亚线性或对数多项式时间算法的设计问题.

4 未来研究方向的思考

通过第 3 节对于大数据计算的复杂性理论和算法的研究进展的分析和总结, 可以得到如下 5 个结论:

(1) 传统的计算模型和 Map-Reduce 计算模型都没有考虑大数据计算的资源受限性, 不能准确地刻画大数据计算, 而且 Map-Reduce 计算模型还存在计算效率低于 PRAM 模型和编程模式单一的问题.

(2) “多项式时间”已经不能作为大数据计算问题易解性的判定标准, “亚线性时间或对数多项式时间”是否可以作为易解性判定的新标准尚需更多更广的实验研究来验证.

(3) 大数据计算复杂性理论研究非常少见, 几乎所有问题都没有解决, 通信复杂性和 I/O 复杂性还没有得到重视.

(4) 大数据计算的算法设计方法不多, 现有的几种方法尚需在更广泛的大数据计算问题求解过程中验证. 值得注意的是, 大数据计算的算法设计方法必须在大量的大数据问题求解实践中凝练、补充和完善.

(5) 大数据查询、挖掘与分析是所有大数据应用的基本工具, 目前已经存在一些研究结果, 但是研究范围还很狭窄, 远远不能满足应用的需要. 主要问题包括: 大数据查询、挖掘与分析问题的固有复杂性几乎无人问津; 多数算法的时间复杂性是多项式时间或更高, 几乎没有考虑亚线性或对数多项式时间算法的设计问题; 算法分析主要集中在时间复杂性, 很少考虑通信复杂性, 没有考虑 I/O 复杂性.

根据上述 5 个结论, 我们认为大数据计算的复杂性理论和算法的未来研究方向包括如下 7 个方面:

(1) 以 Cluster 和具有大规模共享存储器的并行计算机系统 (如正在出现的基于非易失存储器的并行计算系统) 为背景, 建立新的大数据计算模型, 使之能够更准确地刻画大数据计算, 更易于大数据计算复杂性的研究, 具有更高的计算效率.

(2) 研究大数据计算问题易解性判定新标准. 我们已经通过实验研究, 猜想易解性判定新标准可能是亚线性时间或对数多项式时间, 但仍需通过更广泛的实验研究, 验证亚线性或对数多项式时间作为大数据计算问题易解性判定标准的合理性.

(3) 使用大数据计算问题易解性判定新标准, 应用图灵机和新计算模型, 以时间复杂性、空间复杂性、I/O 复杂性和通信复杂性为核心, 研究新的大数据计算的复杂性理论, 包括大数据计算问题的复杂性分类和类之间关系、大数据的确定计算、不确定计算、并行计算、随机计算、近似计算等基础理论.

(4) 使用大数据计算问题易解性判定新标准, 应用图灵机和新计算模型, 研究大数据计算问题的固有计算复杂性和难解性, 包括时间复杂性、空间复杂性、I/O 复杂性和通信复杂性.

(5) 使用大数据计算问题易解性判定新标准, 基于 Cluster、Map-Reduce 和新计算系统, 设计与分析大数据计算问题的高效求解算法, 特别是具有亚线性或对数多项式计算复杂性的算法.

(6) 在设计与分析大数据计算问题求解算法的过程中, 凝炼大数据计算问题高效求解算法的一般设计与分析方法, 探索大数据计算的算法设计和分析方法学.

(7) 应用基础理论研究结果, 基于 Cluster、Map-Reduce 和其他新的并行计算系统, 研制高效的大数据管理、查询、挖掘、分析系统以及各类大数据应用系统.

参考文献

- 1 Shoshani A. Statistical databases: characteristics, problems, and some solutions. In: Proceedings of the 8th International Conference on Very Large Data Bases, Mexico City, 1982. 208–222
- 2 Shoshani A, Olken F, Wong H K T. Characteristics of scientific databases. In: Proceedings of the 10th International Conference on Very Large Data Bases, Singapore, 1984. 147–160
- 3 Shoshani A, Wong H K T. Statistical and scientific database issues. IEEE Trans Softw Eng, 1985, 11: 1040–1047
- 4 Turing A M. On computable numbers, with an application to the entscheidungs problem. Proc London Math Soc, 1936, 2: 230–265
- 5 李建中. 大数据计算的挑战. 见: 香山科学会议, 北京, 2012
- 6 李建中. 大数据计算的基本概念与研究问题. 见: 国家基金委第 89 期双清论坛, 上海, 2014
- 7 Li J Z. Complexity, algorithms and quality of big data intensive computing. In: Proceedings of the 19th International Conference on Database Systems for Advanced Applications, Bali, 2014. 230–265
- 8 李建中. 大数据计算的研究问题和部分解. 见: 第 30 届中国数据库学术会议, 哈尔滨, 2013
- 9 Kleene S C. General recursive functions of natural numbers. MATH ANN, 1936, 112: 727–742
- 10 Post E L. Finite combinatory processes-formulation 1. J Symb Log, 1936, 1: 103–105
- 11 Church A. The Calculi of Lambda-Conversion. Princeton: Princeton University Press, 1951
- 12 Kleene S C. Introduction to Metamathematics. Japan: Ishi Press, 1952
- 13 Hermes H. Enumerability, Decidability, Computability. Berlin: Springer, 1965
- 14 Minsky M L. Computation: Finite and Infinite Machines. Upper Saddle River: Prentice-Hall Inc., 1967
- 15 Davis M. Computability and Unsolvability. New York: McGraw-Hill, 1958
- 16 Shepherdson J C, Sturgis H E. Computability of recursive functions. J ACM, 1963, 10: 217–255
- 17 Cook S A, Reckhow R A. Time-bounded random access machines. In: Proceedings of the 4th Annual ACM Symposium on Theory of Computing, Denver, 1972. 73–80

- 18 Harrison M A. Introduction to Switching and Automata Theory. New York: MacGraw-Hill, 1965
- 19 Savage J E. The Complexity of Computing. New York: Wiley, 1976
- 20 Steven F, James W. Parallelism in random access machines. In: Proceedings of the 10th Annual ACM Symposium on Theory of Computing, San Diego, 1978. 114–118
- 21 van Leeuwen J. Handbook of Theoretical Computer Science (vol. A): Algorithms and Complexity. Cambridge: MIT Press, 1991
- 22 Karloff H, Suri S, Vassilvitskii S. A model of computation for mapreduce. In: Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, Austin, 2010. 938–948
- 23 Sarma A D, Afrati F N, Salihoqlu S, et al. Upper and lower bounds on the cost of a map-reduce computation. Proc VLDB Endowment, 2013, 6: 277–288
- 24 Tao Y F, Lin W Q, Xiao X K. Minimal mapreduce algorithms. In: Proceedings of ACM SIGMOD International Conference on Management of Data, New York, 2013. 529–540
- 25 Afrati F N, Borkar V, Carey M, et al. Map-reduce extensions and recursive queries. In: Proceedings of the 14th International Conference on Extending Database Technology, Uppsala, 2011. 1–8
- 26 Afrati F N, Ullman J D. Optimizing joins in a map-reduce environment. In: Proceedings of the 13th International Conference on Extending Database Technology, Lausanne, 2010. 99–110
- 27 Afrati F N, Ullman J D. Transitive closure and recursive datalog implemented on clusters. In: Proceedings of the 15th International Conference on Extending Database Technology, Berlin, 2012. 132–143
- 28 Afrati F N, Fotakis D, Ullman J D. Enumerating subgraph instances using map-reduce. In: Proceedings of IEEE 29th International Conference on Data Engineering (ICDE), Brisbane, 2013. 62–73
- 29 Afrati F N, Sarma A D, Menestrina D, et al. Fuzzy joins using mapreduce. In: Proceedings of IEEE 28th International Conference on Data Engineering (ICDE), Washington, 2012. 498–509
- 30 Afrati F N, Koutris P, Suci D, et al. Parallel skyline queries. Theory Comput Syst, 2015, 57: 1008–1037
- 31 Beame P, Koutris P, Suci D. Communication steps for parallel query processing. In: Proceedings of the 32nd ACM Symposium on Principles of Database Systems, New York, 2013. 273–284
- 32 Fan W F, Geerts F, Neven F. Making queries tractable on big data with preprocessing. Proc VLDB Endowment, 2013, 6: 685–696
- 33 Dean J, Ghemawat S. Mapreduce: simplified data processing on large clusters. Commun ACM, 2008, 51: 107–113
- 34 Wang J B, Wu S, Gao H, et al. Indexing multi-dimensional data in a cloud system. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Indianapolis, 2010. 591–602
- 35 Alper O, Mirek R. Processing theta-joins using mapreduce. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, Athens, 2011. 949–960
- 36 Kim Y, Shim K. Parallel top-k similarity join algorithms using mapreduce. In: Proceedings of IEEE 28th International Conference on Data Engineering (ICDE), Washington, 2012. 510–521
- 37 Lu W, Shen Y Y, Chen S, et al. Efficient processing of k nearest neighbor joins using mapreduce. Proc VLDB Endowment, 2012, 5: 1016–1027
- 38 Zhang X F, Chen L, Wang M. Efficient multi-way theta-join processing using mapreduce. Proc VLDB Endowment, 2012, 5: 1184–1195
- 39 Chen G, Vo H T, Wu S, et al. A framework for supporting dbms-like indexes in the cloud. Proc VLDB Endowment, 2011, 4: 702–713
- 40 Miliaraki I, Berberich K, Gemulla R, et al. Mind the gap: large-scale frequent sequence mining. In: Proceedings of ACM SIGMOD International Conference on Management of Data, New York, 2013. 797–808
- 41 Amol G, Rajasekar K, Edwin P D P, et al. Systemml: declarative machine learning on mapreduce. In: Proceedings of IEEE 27th International Conference on Data Engineering (ICDE), Hannover, 2011. 231–242
- 42 Zhang Z J, Shu H, Chong Z H, et al. C-cube: elastic continuous clustering in the cloud. In: Proceedings of IEEE 29th International Conference on Data Engineering (ICDE), Brisbane, 2013. 577–588
- 43 Pansare N, Borkar V R, Jermaine C, et al. Online aggregation for large mapreduce jobs. Proc VLDB Endowment, 2011, 4: 1135–1145
- 44 Grover R, Carey M J. Extending map-reduce for efficient predicate-based sampling. In: Proceedings of IEEE 28th International Conference on Data Engineering (ICDE), Washington, 2012. 486–497

- 45 Jestes J, Yi K, Li F F. Building wavelet histograms on large data in mapreduce. *Proc VLDB Endowment*, 2011, 5: 109–120
- 46 Huang B T, Babu S, Yang J. Cumulon: optimizing statistical data analysis in the cloud. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, New York, 2013. 1–12
- 47 Zhang C, Ré C. Towards high-throughput gibbs sampling at scale: a study across storage managers. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, New York, 2013. 397–408
- 48 Zheng Y, Jestes J, Phillips J M, et al. Quality and efficiency for kernel density estimates in large data. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, New York, 2013. 433–444
- 49 Wong H K T, Li J Z, Olken F, et al. Bit transposition for very large scientific and statistical databases. *Algorithmica*, 1986, 1: 289–309
- 50 Li J Z, Rotem D, Wong H K T. A new compression method with fast searching on large databases. In: *Proceedings of the 13th International Conference on Very Large Data Bases*, Brighton, 1987. 311–318
- 51 Li J Z, Harry K T W, Doron R. Batched interpolation searching on databases. In: *Proceedings of the IEEE 3rd International Conference on Data Engineering (ICDE)*, Los Angeles, 1987. 79–97
- 52 Wong H K T, Li J Z, Wong H K T, et al. Transposition algorithms on very large compressed databases. In: *Proceedings of the 12th International Conference on Very Large Data Bases*, Kyoto, 1986. 304–311
- 53 Li J Z, Rotem D, Srivastava J. Aggregation algorithms for very large compressed data warehouses. In: *Proceedings of the 25th International Conference on Very Large Data Bases*, Edinburgh, 1999. 651–662
- 54 Wu W L, Gao H, Li J Z. New algorithm for computing cube on very large compressed data sets. *IEEE Trans Knowl Data Eng*, 2006, 18: 1667–1680
- 55 Fan W F, Li J Z, Wang X, et al. Query preserving graph compression. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Scottsdale, 2012. 157–168
- 56 Zhang S, Li J Z, Gao H, et al. A novel approach for efficient supergraph query processing on graph databases. In: *Proceedings of the 12th International Conference on Extending Database Technology*, Petersburg, 2009. 204–215
- 57 Cheng S Y, Li J Z. Sampling based (ϵ , δ)-approximate aggregation algorithm in sensor networks. In: *Proceedings of the 29th IEEE International Conference on Distributed Computing Systems (ICDCS)*, Montreal, 2009. 273–280
- 58 Li J Z, Cheng S Y. (ϵ , δ)-approximate aggregation algorithms in dynamic sensor networks. *IEEE Trans Parall Distrib Syst*, 2012, 23: 385–396
- 59 Cheng S Y, Li J Z, Ren Q Q, et al. Bernoulli sampling based (ϵ , δ)-approximate aggregation in large-scale sensor networks. In: *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, San Diego, 2010. 1181–1189
- 60 Cheng S Y, Cai Z P, Li J Z, et al. Drawing dominant dataset from big sensory data in wireless sensor networks. In: *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, Kowloon, 2015. 147–152
- 61 Liu Y, Li J Z, Gao H, et al. Enabling ϵ -approximate querying in sensor networks. *Proc VLDB Endowment*, 2009, 2: 169–180
- 62 Gao J, Li J Z, Zhang Z G, et al. An incremental data stream clustering algorithm based on dense units detection. In: *Proceedings of the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD)*, Hanoi, 2005. 420–425
- 63 Fan W F, Li J Z, Luo J Z, et al. Incremental graph pattern matching. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Athens, 2011. 925–936
- 64 Fan W F, Li J Z, Tang N, et al. Incremental detection of inconsistencies in distributed data. In: *Proceedings of IEEE 28th International Conference on Data Engineering (ICDE)*, Washington, 2012. 318–329
- 65 Brodal G S, Tsakalidis K, Sioutas S, et al. Fully persistent b-trees. In: *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms*, Kyoto, 2012. 602–614
- 66 Ulrich M, Norbert Z. I/O-efficient shortest path algorithms for undirected graphs with random or bounded edge lengths. *ACM Trans Algorithms*, 2012, 8: 22
- 67 Rasmussen C K, Tao Y F, Tsakalidis K, et al. I/O-efficient planar range skyline and attrition priority queues. In: *Proceedings of the 32nd ACM Symposium on Principles of Database Systems*, New York, 2013. 103–114
- 68 Huang J W, Venkatraman K, Abadi D J. Query optimization of distributed pattern matching. In: *Proceedings of*

- IEEE 30th International Conference on Data Engineering (ICDE), Chicago, 2014. 64–75
- 69 Deng D, Li G L, Hao S, et al. Massjoin: a mapreduce-based method for scalable string similarity joins. In: Proceedings of IEEE 30th International Conference on Data Engineering (ICDE), Chicago, 2014. 340–351
- 70 Minsik C, Daniel B, Rajesh B, et al. Paradis: an efficient parallel algorithm for in-place radix sort. Proc VLDB Endowment, 2015, 8: 1518–1529
- 71 Shumo C, Magdalena B, Dan S. From theory to practice: efficient join query evaluation in a parallel database system. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Melbourne, 2015. 63–78
- 72 Jennie D, Olga P, Leilani B, et al. Skew-aware join optimization for array databases. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Melbourne, 2015. 123–135
- 73 Orestis P, Rajkumar S, Kenneth A R. Track join: distributed joins with minimal network traffic. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Snowbird, 2014. 1483–1494
- 74 Amr E, Venkatesh R, Mohamed A S, et al. Optimization of common table expressions in mpp database systems. Proc VLDB Endowment, 2015, 8: 1704–1715
- 75 Han X X, Li J Z, Wang J B, et al. Tjje: an efficient algorithm for top-k join on massive data. Inf Sci, 2013, 222: 362–383
- 76 Han X X, Li J Z, Yang D H, et al. Efficient skyline computation on big data. IEEE Trans Knowl Data Eng, 2013, 25: 2521–2535
- 77 Khayyat Z, Lucia W, Singh M, et al. Lightning fast and space efficient inequality joins. Proc VLDB Endowment, 2015, 8: 2074–2085
- 78 Fan W F, Wang X, Wu Y H. Querying big graphs within bounded resources. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Snowbird, 2014. 301–312
- 79 Fan W F, Geerts F, Cao Y, et al. Querying big data by accessing small data. In: Proceedings of the 34th ACM Symposium on Principles of Database Systems, Melbourne, 2015. 173–184
- 80 Fan W F, Geerts F, Libkin L. On scale independence for querying big data. In: Proceedings of the 33rd ACM Symposium on Principles of Database Systems, Snowbird, 2014. 51–62
- 81 Cao Y, Fan W F, Huai J P, et al. Making pattern queries bounded in big graphs. In: Proceedings of the IEEE 31th International Conference on Data Engineering (ICDE), Seoul, 2015. 161–172
- 82 Fan W F, Wang X, Wu Y H, et al. Distributed graph simulation: impossibility and possibility. Proc VLDB Endowment, 2014, 7: 1083–1094
- 83 Fan W F, Wang X, Wu Y H, et al. Association rules with graph patterns. Proc VLDB Endowment, 2015, 8: 1502–1513
- 84 Huang J W, Abadi D J, Ren K. Scalable sparql querying of large rdf graphs. Proc VLDB Endowment, 2011, 4: 1123–1134
- 85 Zhang X F, Chen L, Tong Y X, et al. Eagre: towards scalable I/O efficient sparql query evaluation on the cloud. In: Proceedings of IEEE 29th International Conference on Data Engineering (ICDE), Brisbane, 2013. 565–576
- 86 Zeng K, Yang J C, Wang H X, et al. A distributed graph engine for web scale rdf data. Proc VLDB Endowment, 2013, 6: 265–276
- 87 Yuan P P, Liu P, Wu B W, et al. Triplebit: a fast and compact system for large scale rdf data. Proc VLDB Endowment, 2013, 6: 517–528
- 88 Zheng W G, Zou L, Feng Y S, et al. Efficient simrank-based similarity join over large graphs. Proc VLDB Endowment, 2013, 6: 493–504
- 89 Mondal J, Deshpande A. Managing large dynamic graphs efficiently. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Scottsdale, 2012. 145–156
- 90 Yang S Q, Yan X F, Zong B, et al. Towards effective partition management for large graphs. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Scottsdale, 2012. 517–528
- 91 Zhao P X, Aggarwal C C, Wang M. Gsketch: on query estimation in graph streams. Proc VLDB Endowment, 2011, 5: 193–204
- 92 Shao Y X, Cui B, Chen L, et al. Parallel subgraph listing in a large-scale graph. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Snowbird, 2014. 625–636
- 93 Li Z G, Fang Y X, Liu Q, et al. Walking in the cloud: parallel simrank at scale. Proc VLDB Endowment, 2015, 9:

- 24–35
- 94 Zhu Y Y, Yu J X, Qin L. Leveraging graph dimensions in online graph search. *Proc VLDB Endowment*, 2014, 8: 85–96
- 95 Qi Z C, Xiao Y H, Shao B, et al. Toward a distance oracle for billion-node graphs. *Proc VLDB Endowment*, 2013, 7: 61–72
- 96 Qin L, Yu J X, Chang L J, et al. Scalable big graph processing in mapreduce. In: *Proceedings of ACM SIGMOD International Conference on Management of Data, Snowbird*, 2014. 827–838
- 97 Levin R, Kanza Y. Stratified-sampling over social networks using mapreduce. In: *Proceedings of ACM SIGMOD International Conference on Management of Data, Snowbird*, 2014. 863–874
- 98 Zhu A D, Lin W Q, Wang S B, et al. Reachability queries on large dynamic graphs. In: *Proceedings of ACM SIGMOD International Conference on Management of Data, Snowbird*, 2014. 1323–1334
- 99 Gao J, Jin R M, Zhou J S, et al. Relational approach for shortest path discovery over large graphs. *Proc VLDB Endowment*, 2011, 5: 358–369
- 100 Sun Z, Wang H Z, Wang H X, et al. Efficient subgraph matching on billion node graphs. *Proc VLDB Endowment*, 2012, 5: 788–799
- 101 Jin R M, Ruan N, Dey S, et al. Scarab: scaling reachability computation on large graphs. In: *Proceedings of ACM SIGMOD International Conference on Management of Data, Scottsdale*, 2012. 169–180
- 102 Chen H Q, Ku W, Wang H X, et al. Linkprobe: probabilistic inference on large-scale social networks. In: *Proceedings of IEEE 29th International Conference on Data Engineering (ICDE)*, Brisbane, 2013. 290–301
- 103 Xiang J, Guo C, Abounaga A. Scalable maximum clique computation using mapreduce. In: *Proceedings of IEEE 29th International Conference on Data Engineering (ICDE)*, Brisbane, 2013. 74–85
- 104 Zhang Z W, Yu J X, Qin L, et al. I/O efficient: computing sccs in massive graphs. *VLDB J*, 2015, 24: 245–270
- 105 Shun J, Tangwongsan K. Multicore triangle computations without tuning. In: *Proceedings of IEEE 31st International Conference on Data Engineering (ICDE)*, Seoul, 2015. 149–160
- 106 Chitnis L, Das S A, Machanavajjhala A, et al. Finding connected components in map-reduce in logarithmic rounds. In: *Proceedings of IEEE 29th International Conference on Data Engineering (ICDE)*, Brisbane, 2013. 50–61
- 107 Beedkar K, Gemulla R. Lash: large-scale sequence mining with hierarchies. In: *Proceedings of ACM SIGMOD International Conference on Management of Data, Melbourne*, 2015. 491–503
- 108 Buehrer G, De O R L, Fuhry D, et al. Towards a parameter-free and parallel itemset mining algorithm in linearithmic time. In: *Proceedings of IEEE 31st International Conference on Data Engineering (ICDE)*, Seoul, 2015. 1071–1082
- 109 Schelter S, Soto J, Markl V, et al. Efficient sample generation for scalable meta learning. In: *Proceedings of IEEE 31st International Conference on Data Engineering (ICDE)*, Seoul, 2015. 1191–1202
- 110 Riondato M, Debrabant J A, Fonseca R, et al. Parma: a parallel randomized algorithm for approximate association rules mining in mapreduce. In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*, Maui, 2012. 85–94
- 111 Riondato M, Upfal E. Mining frequent itemsets through progressive sampling with rademacher averages. In: *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney*, 2015. 1005–1014
- 112 Wang P H, Lui J C S, Towsley D. Minfer: inferring motif statistics from sampled edges. In: *Proceedings of IEEE 32nd International Conference on Data Engineering (ICDE)*, Helsinki, 2016. 1050–1061
- 113 Liang Y Y, Xie B, Woodruff D, et al. Communication efficient distributed kernel principal component analysis. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco*, 2016, in press
- 114 Zhang A, Gu Q Q. Accelerated stochastic block coordinate descent with optimal sampling. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco*, 2016, in press
- 115 Yang Y, Chen J F, Zhu J. Distributing the stochastic gradient sampler for large-scale lda. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco*, 2016, in press
- 116 Kang U, Chau D H, Faloutsos C. Mining large graphs: algorithms, inference, and discoveries. In: *Proceedings of*

- IEEE 27th International Conference on Data Engineering (ICDE), Hannover, 2011. 243–254
- 117 Morales G D F, Gionis A, Sozio M. Social content matching in mapreduce. *Proc VLDB Endowment*, 2011, 4: 460–469
- 118 Bahman B, Kaushik C, Dong X. Fast personalized pagerank on mapreduce. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Athens, 2011. 973–984
- 119 Chu L Y, Wang S H, Liu S Y, et al. Alid: scalable dominant cluster detection. *Proc VLDB Endowment*, 2015, 8: 826–837
- 120 Lin W Q, Xiao X K, Ghinita G. Large-scale frequent subgraph mining in mapreduce. In: *Proceedings of IEEE 30th International Conference on Data Engineering (ICDE)*, Chicago, 2014. 844–855
- 121 Alvanaki F, Michel S. Tracking set correlations at large scale. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Snowbird, 2014. 1507–1518
- 122 Li F, Ozsu M T, Chen G, et al. R-store: a scalable distributed system for supporting real-time analytics. In: *Proceedings of IEEE 30th International Conference on Data Engineering (ICDE)*, Chicago, 2014. 40–51
- 123 Scholer F, Williams H E, Yiannis J, et al. Compression of inverted indexes for fast query evaluation. In: *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*, Tampere, 2002. 222–229
- 124 Sihem A, Theodore J. Optimizing queries on compressed bitmaps. In: *Proceedings of the 26th International Conference on Very Large Data Bases*, Cairo, 2010. 329–338
- 125 Yan Y, Zhang J X, Huang B J, et al. Distributed outlier detection using compressive sensing. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Melbourne, 2015. 3–16
- 126 Chang L J, Yu J X, Qin L, et al. Efficiently computing k-edge connected components via graph decomposition. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, New York, 2013. 205–216
- 127 Song C Y, Ge T J, Chen C, et al. Event pattern matching over graph streams. *Proc VLDB Endowment*, 2014, 8: 413–424
- 128 Shiokawa H, Fujiwara Y, Onizuka M. Scan++: efficient algorithm for finding clusters, hubs and outliers on large-scale graphs. *Proc VLDB Endowment*, 2015, 8: 1178–1189
- 129 Liu Y, Lu J H, Yang H, et al. Towards maximum independent sets on massive graphs. *Proc VLDB Endowment*, 2015, 8: 2122–2133
- 130 Min F, Narayanan S, Hector G M, et al. Computing iceberg queries efficiently. In: *Proceedings of the 24th International Conference on Very Large Data Bases*, New York, 1998. 299–310
- 131 Li N, Guan Z Y, Ren L J, et al. Giceberg: towards iceberg analysis in large graphs. In: *Proceedings of IEEE 29th International Conference on Data Engineering (ICDE)*, Brisbane, 2013. 1021–1032
- 132 Cui W Y, Xiao Y H, Wang H X, et al. Local search of communities in large graphs. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Snowbird, 2014. 991–1002
- 133 Huang X, Cheng H, Qin L, et al. Querying k-truss community in large and dynamic graphs. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Snowbird, 2014. 1311–1322
- 134 Elgamal T, Yabandeh M, Aboulnaga A, et al. Spca: scalable principal component analysis for big data on distributed platforms. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Melbourne, 2015. 79–91
- 135 Yu L L, Shao Y X, Cui B. Exploiting matrix dependency for efficient distributed matrix computation. In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Melbourne, 2015. 93–105
- 136 Ghashami M, Phillips J M, Li F F. Continuous matrix approximation on distributed data. *Proc VLDB Endowment*, 2014, 7: 809–820
- 137 Papapetrou O, Garofalakis M. Continuous fragmented skylines over distributed streams. In: *Proceedings of IEEE 30th International Conference on Data Engineering (ICDE)*, Chicago, 2014. 124–135
- 138 Cao L, Yang D, Wang Q Y, et al. Scalable distance-based outlier detection over high-volume data streams. In: *Proceedings of IEEE 30th International Conference on Data Engineering (ICDE)*, Chicago, 2014. 76–87
- 139 Aggarwal C C, Yu P S. On historical diagnosis of sensor streams. In: *Proceedings of IEEE 31st International Conference on Data Engineering (ICDE)*, Seoul, 2015. 185–194
- 140 Sadoghi M, Jacobsen H. Adaptive parallel compressed event matching. In: *Proceedings of IEEE 30th International Conference on Data Engineering (ICDE)*, Chicago, 2014. 364–375

- 141 Reynold S X, Josh R, Matei Z, et al. Shark: SQL and rich analytics at scale. In: Proceedings of ACM SIGMOD International Conference on Management of Data, New York, 2013. 13–24
- 142 Tsytsarau M, Amer-yahia S, Palpanas T. Efficient sentiment correlation for large-scale demographics. In: Proceedings of ACM SIGMOD International Conference on Management of Data, New York, 2013. 253–264
- 143 Gatterbauer W, Nnemann S, Koutra D, et al. Linearized and single-pass belief propagation. Proc VLDB Endowment, 2014, 8: 581–592
- 144 Park Y, Min J K, Shim K. Parallel computation of skyline and reverse skyline queries using mapreduce. Proc VLDB Endowment, 2013, 6: 2002–2013

Research progress in the complexity theory and algorithms of big-data computation

Jianzhong LI^{1*} & Yingshu LI²

1 *School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China;*

2 *Department of Computer Science, Georgia State University, Atlanta GA 30302-3965, USA*

*E-mail: lijzh@hit.edu.cn

Abstract With the explosive growth of available data in recent years, big-data research has attracted much attention from both academic and industrial researchers, and many significant advances have been achieved. Nevertheless, the fundamental research results are far from the actual needs, a number of key issues remain unresolved, a complete theory of big-data computation needs to be established, and considerable work remains to be accomplished. This paper focuses on the theoretical aspects of big-data research, especially the research progress of the complexity theory and algorithms of big-data computation. First, big-data computation is formally defined. Then, six challenges and ten research issues of big data are discussed. Next, the results of a survey on the research progress of complexity theory and algorithms of big-data computation are given. Finally, comments on the fundamental research results and the future theoretical research issues of big data are presented and discussed.

Keywords big data, big data computation, complexity of big data computation, algorithm of big data computation



Jianzhong LI was born in 1950. He graduated from Heilongjiang University, Harbin, China, in 1975. Currently, he is a professor at the Harbin Institute of Technology. His research interests include big-data computation and wireless sensor networks.



Yingshu LI was born in 1978. She received her Ph.D. degree in Computer Science from the University of Minnesota, USA, in 2005. Currently, she is an associate professor at Georgia State University. Her research interests include databases and wireless networks.