面向政府治理大数据的高性能计算系统

吴维刚[1]，常亮[2]，任江涛[1]，古天龙[2]
1. 中山大学数据科学与计算机学院，广东 广州 510006；
2. 桂林电子科技大学计算机与信息安全学院，广西 桂林 541004

摘要
大数据处理系统是未来社会的基础设施之一。政府治理场景下的大数据处理任务具有多域异构、多主体等特点，因此需要针对性地进行研究设计。从应用需求出发，分析各类政府治理场景对大数据处理技术提出的挑战，梳理大数据分布并行处理的关键技术，包括数据存储管理、计算平台、关键算法等，调研总结相关技术的研究现状，并提出面向政府治理大数据的高性能计算系统的技术框架，分析讨论不同技术路线的优劣。最后展望相关技术的未来发展趋势。

关键词
大数据处理；政府治理；分布式计算；计算框架；资源管理

中图分类号：TP316 文献标识码：A doi: 10.11959/j.issn.2096-0271.2020013

High performance big data computing systems for government governance

WU Weigang[1], CHANG Liang[2], REN Jiangtao[1], GU Tianlong[2]
1. School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China
2. School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China

Abstract
The big data processing system would be one of the major infrastructures of future society. The characteristics and requirements of government governance applications, including multi-domain, multi-entity, etc., bring new challenges to big processing platform, and new design is desirable. Based on the requirements of applications, the challenges in big data processing for government governance scenarios were analyzed, and key techniques in distributed parallel processing for public governance were discussed, including data storage, computing platform, and key algorithms. The state-of-art techniques were investigated and reviewed, and a candidate framework for government governance data process system was proposed, the pros and cons of different technical methodologies were also discussed. Finally, the open problems and future directions were also discussed.

Key words
big data processing, government governance, distributed computing, computing framework, resource management

1 引言

随着互联网、物联网、云计算等信息与通信技术(information and communications technology, ICT)的迅猛发展，大数据时代已经来临。政府拥有和管理了规模巨大的政务大数据，包括公安、交通、医疗卫生、民政、就业等因开展政府工作而产生和采集的海量数据以及因管理服务需求而采集的外部与政务有关的大数据，如互联网舆情数据、电信网络数据等。大数据已经渗透到工业和商业领域的各个方面，成为影响生产的重要因素。政府治理活动迫切需要大数据技术的支撑和保障。在大数据条件下，数据驱动的“精准治理体系”“智慧决策体系”“阳光权力平台”将逐渐成为现实。

目前，国内外学者对政府治理大数据的技术研究和应用做了大量工作。但是，政府治理大数据的技术整体上还处在非常初始的阶段。现有的应用大多是针对特定、单一功能进行设计实现的，还缺乏综合性应用。在政务大数据分析处理系统方面，大多基于一般的服务器集群并未考虑利用已经大量建设和部署的超级计算系统。本文将首先介绍大数据应用在政府治理领域遇到的挑战，然后从大数据的存储与管理平台、政府治理大数据的分析处理平台出发，介绍政务大数据关键技术和算法，梳理相关技术的研究现状，并提出基于高性能超级计算平台的政务大数据处理系统。

2 应用情况

大数据在政府中的应用十分广泛[1-2]，本节从政策效果评估预测、网络舆情分析、社会信用风险评估以及智慧城市构建4个方面介绍政务大数据在政府治理中的典型应用场景以及具有代表性的应用实例。

在政策效果评估预测领域，韩国庆北大学的Jun等人[3]使用文本大数据管理解决方案Textom对地方政府的Government 3.0项目进行了评估。首先，通过Textom对韩国两大门户网站Naver和Daum上关于庆尚北道的数据进行了收集，包括新闻、文档、照片等。然后对收集的数据进行语义网络分析，得出对庆尚北道Government 3.0项目的结构化理解，同时为该项目提供了一个全面的评估。

在网络舆情分析方面，国内外已有众多成果，其中有代表性的包括国外的Twelvefold、Buzz、Metrics、Reputation Defender、Cision以及国内的人大方正、Rank、Goonie、军犬、麦知讯等舆情监控系统。大数据环境下的舆情分析主要包括信息采集、热点发现、热点评估与跟踪、分析处理4个方面[4]。其中，信息采集包含数据爬取、存储及清洗。可通过网络爬虫、网站应用程序接口(application programming interface, API)获得所需数据；对于数据存储来说，当前有海量非结构化数据的分布式文件存储系统、海量半结构化数据的NoSQL数据库和海量结构化的分布式并行数据库系统3种大数据存储技术；数据清洗则是删除无效的网页数据和重复的文本数据。热点发现强调对新信息的发现和对特定热点的关注，通过聚类将信息汇总，并自动跟踪新闻事件，提供事件发展的轨迹[5-6]，其常用的技术有Single-pass聚类算法、*K*-means聚类算法、KNN算法、支持向量机(SVM)、SOM神经网络聚类算法等。热点评估与跟踪关注的是如何根据热点事件中公众的情感和行为反应对舆情进行等级评估并设立相应的预警阈值。主要手段为词频统计和情感分类。词频统计是指对网络调查数据、文章关键词、浏览统计数据等进行采集分析及评

估，对文本量大的结构化数据处理效果较好。情感分析则依赖于2类关键技术：基于概率论、信息论的分类算法和基于机器学习的分类算法。当前主流的算法为朴素贝叶斯算法和KNN算法。分析处理主要是根据分析的舆情等级及相应标准采取对应的控制与引导策略，常用的分类技术有贝叶斯分类技术、神经网络和SVM。

在社会信用风险评估方面，比较有代表性的应用包括国外的Big Data Scoring和国内的“信用天眼”。Big Data Scoring能够给银行、P2P贷款平台、小额信贷提供商和租赁公司等贷方提供易于集成的、基于云的服务，通过大数据分析提高贷款质量和接受率。该系统从贷款申请人的社交媒体、Google检索、IP地址等网络数据源收集数据，并将其与申请人的网络行为关联，在几秒内就可以准确预测潜在的客户付款行为，帮助贷方做出更有利的信用决策。“信用天眼”是由九次方大数据信息集团有限公司研发的社会信用大数据平台，该平台通过大数据分析技术建立信用模型，实现信用主体的综合信用评价，生成信用报告，并对具有信用风险的主体进行预警。目前，“信用天眼”主要包括“一网三库一平台”。其中，“一网”是指信用官方网站；“三库”是指归集、完善和整合各行业、各领域的信用信息建设成果，依托统一的社会信用代码，分别建立企业、个人、非企业法人（政府机关、事业单位、社会团体等）3个社会信用信息基础数据库；“一平台”是指利用大数据、云计算等技术，将三库信息进行融合，建立社会信用信息交换共享平台。

此外，在智慧城市构建方面，Rathore等人[7]提出了一个基于物联网设备的4层模型，根据该模型产生的大数据构建智慧城市。在巴西里约热内卢，政府与IBM公司合作成立了一个仪表系统[8]，将从30个代理处获得的包括交通、公共服务、紧急服务、天气摘要以及员工和民众提交的各种信息整合到一个分析中心。在这里，巨量的实时信息被整合、分析、可视化，这些信息被用于了解城市各方面的状态，构建模型预测城市的改变，同时也被用于预防洪水等灾害。一个具体的例子是，警方在事故现场可以通过该平台查看救护车的派遣情况，并上传现场信息。

3 技术需求与挑战

利用大数据分析处理技术实现政府治理大数据的有效管理和利用，并通过相应的应用服务于政府治理需求，仍然面临很多的挑战。

3.1 政府治理大数据的多源、异质、异构特性

建立政府治理大数据存储与管理基础设施是开展基于大数据的政府治理的基础。政府治理大数据涵盖政府各部门、企事业单位、居民等方面的各类数据，主要具有如下特征。

- 由于涉及的数据范围广、数量多，数据呈现多源、异质、异构等特点。
- 由于拥有丰富数据的政府部门彼此之间协调合作不足，“信息孤岛”现象普遍存在。
- 社交媒体、金融、电商、医疗、教育、交通等行业的数据正对政府治理产生日益重要的影响，而这些数据并不完全由政府自身掌握。

上述这些特点对大数据的存储、管理、融合都提出了新要求。

此外，政府治理大数据呈现多样化的发展趋势，其不仅涉及众多数据库中存储的结构化数据，还涉及大量的半结构化和非结构化数据，例如政府治理者可以从传感器、

卫星、社交媒体、移动通信、电子邮件、无线射频识别设备等新兴途径中获得海量的、类型多样的数据，而这些数据集通常是以原始格式发布的，缺乏编码一致性。

由此可见，在推动政府治理大数据应用的过程中，不仅需要推动政府之间的数据共享与业务协同，打破部门孤岛，推进数据的集成，并逐步整合政府外部的数据资源，消弭“数据孤岛”之间的数据表示和数据语义隔阂；更需要针对数据的多源性、异构性、异质性给大数据存储管理带来的新挑战，在确保数据可信、安全与隐私的基础上，实现数据的高效访问和融合，进而构建大数据集成与共享基础设施，以满足政府治理的大数据存储、管理与融合需求。

3.2 政府治理大数据的应用的复杂性、多样性

政府治理大数据的分析处理需要兼顾多处理模式的计算框架。与政府治理相关的大数据具有明显的多源性和多样性，而政府治理活动本身则呈现出高频实时、深度定制化、全周期沉浸式交互、跨组织数据整合、多主体决策等特征。数据和应用的多样性、复杂性使得政府治理大数据处理框架需要同时兼顾不同的处理模式。例如，治安监控视频的分析与识别属于计算密集型处理，互联网论坛文本数据的挖掘分析属于输入/输出（input/output，I/O）密集型处理，政府开放数据服务需要支持大量并发用户的高吞吐量处理模式，而有些处理任务则需要结合多种不同的处理模式。这样的数据特性和应用需求必然要求政府治理大数据处理系统要多方兼顾，实现不同处理模式的共存、融合。因此，支持多处理模式的计算框架是政府治理大数据处理系统和应用的迫切要求。

现有的并行与分布式处理框架通常是为单一的计算处理模式设计的，还不能兼顾不同的处理模式。为了运行一个综合性的、包含多种处理模式的大数据应用，不同模式的计算任务要提交到不同处理模式的多个平台上执行。这必然带来由任务切换、数据通信、资源管理等多方面因素导致的开销和成本，严重影响执行效率，造成资源浪费。因此，在大数据处理框架方面，需要进行融合设计，实现综合计算效率的均衡。

然而，不同处理模式的融合设计是一个富有挑战性的任务。现有的分布式并行计算系统大概可以分为面向高性能计算的超级计算框架和面向海量数据处理的分布式集群框架两大类。超级计算机主要采用信息传递接口（message passing interface，MPI）编程模型，计算框架由一个或多个彼此通过库函数进行消息收发通信的进程组成。超级计算平台的应用针对具体需求进行优化，包括在计算模型、负载均衡策略和通信等多方面进行优化设计，支持复杂的并行应用。而分布式集群框架则基于MapReduce的易并行（embarrassingly parallel）技术进行数据处理，数据和任务分割、网络通信交给框架实现，简单易用，可扩展性和可靠性高，但是由于其并行模式相对简单，无法处理复杂的并行性。

现有的2类分布并行计算框架在系统结构、编程模型及运行环境方面都有很大不同，如何面向政府治理大数据的处理需求进行融合，实现统一的高性能海量数据处理框架是一个重要问题。

4 关键技术

4.1 大数据的存储与管理技术

面向政府治理大数据的存储与管理是

“数据开放”和“数据分析”的基础支撑技术。政府治理大数据具有多源、异构、异质特征，面向政府治理的应用对数据访问的需求具有多样性特征。大数据存储与管理是政府治理大数据处理的前提，是建立高效准确的政府治理且进行规模化应用的基础。政府治理可以基于高性能计算机系统的计算架构特性特征、存储与I/O优势等，从大数据的存储、管理、融合3个角度深入研究政府治理大数据存储与管理的核心技术，以方便上层应用获取数据。具体技术包括以下内容。

（1）面向政府治理大数据的混合式存储系统

一方面，不同的数据对存储系统有不同的要求。例如，视频监控数据采用文件方式保存，经济运行指标数据采用传统的关系数据库存储，各类案件的记录描述可能采用文本形式存储，而一些行为信息可能采用NoSQL的键值对存储。另一方面，不同的技术框架采用的存储方式和系统也有差别。如MPI的高性能计算机系统框架可能把数据存储在SQL数据库和并行文件系统中，而MapReduce框架则基于Hadoop分布式文件系统（Hadoop distributed file system，HDFS）、NoSQL数据库存储文件。为此，需要针对高性能计算机系统的存储特性，研究能够整合封装不同存储模型的存储管理中间件，实现不同存储技术、存储方式的融合。

（2）面向政府治理的大规模多样性数据获取技术

政府治理大数据处理需要高通量、可伸缩、负载自均衡的分布式数据采集方法。面向政府治理的数据采集是一个实时、持续性的过程，其面向的采集对象具有多样性、分布广泛性和数据生成速度不稳定性的特点，因此需要具有高通量、可伸缩特性的分布式数据采集方法，并且能够支持数据采集负载的自均衡，充分开发高性能计算机系统的硬件性能，满足大规模多样性数据的实时采集需求。

（3）面向政府治理大数据的数据共享访问方法

政府治理大数据处理需要基于多级分布式索引结构和多粒度的数据共享机制。政府治理的各项分析应用需要多类数据协同工作，因此需要考虑数据联动访问及高并发的数据请求。而且，由于分析目标不同，应用对目标数据的请求粒度也不同，所以需要基于存储和计算特性设计支持高并发、多粒度读操作的分布式索引结构，支持数据联动访问，实现政府治理大数据的高并发、柔性粒度共享。

（4）面向政府治理大数据的数据质量保证技术

政府治理大数据处理需要建立针对政务数据的元数据信息构建及维护机制。政务数据覆盖了政府治理数据的所有基础信息，具有多源异构、关系松散、数据冗余和不一致性的特点。而政府治理需要进行数据联动访问，因此需要从语义层面研究数据源之间及数据源内部的元数据信息构建及维护方法，进而基于数据关联和数据冗余，设计数据约束和数据演化推理方法，修正多源异构数据之间的数据不一致性，保证上层分析应用高质量的数据联动访问。

4.2 大数据的分析处理技术

由于数据的复杂多样性，在大数据处理的整个过程中，应用负载也表现出多种模式，因此需要考虑不同的计算模式需求及高性能高数据吞吐的处理过程、关键算法的计算过程的并行优化等。为了处理如此复杂多样的数据和应用，需要对分布并行计算平台进行创新研究设计。具体包括2个方面的研究内容：大数据处理框架与高性能计算框架的融合以及基于融合计算框架的政府治理

大数据分析处理的关键算法，特别是对机器学习和图计算关键算法的并行优化。

（1）融合大数据处理模式与高性能计算模式的混合计算框架

针对政府治理大数据的多种应用，基于高性能计算机系统，研究大数据处理与高性能计算不同计算模式的融合框架，支持map/reduce和MPI+OpenMPI的混合计算。为此，需要研究2种框架的融合方式：混合式应用程序设计方法、混合式计算任务管理和调度机制。

在计算框架的融合方式方面，需要采用合适的机制和方法，使得一个应用能够将不同的任务提交到不同的框架上计算，这样才能将政府治理大数据分析处理平台作为一个整体来使用。相应地，需要采用适宜的编程方法将MPI程序和MapReduce程序进行融合，并将其作为一个整体提交到政府治理大数据分析处理平台。

（2）基于融合计算框架的政府治理大数据分析处理关键算法

虽然政府治理大数据在数据特征、应用特性、计算模式等方面具有明显的多样性和复杂性，其所需要的数据分析处理模型和算法却具有明显的共性。机器学习和图计算处于政府治理大数据分析处理计算任务的核心地位，是研究设计政府治理大数据应用的关键部分，其中，深度学习已经成为大数据处理的共性关键技术，在各个应用领域都有重要的基础作用。在政府治理大数据分析处理中，深度学习也将扮演极重要的角色。

虽然在机器学习方面，特别是深度学习和图计算方面已经有不少的并行优化研究和相应的并行化算法、并行化工具库，但是基于高性能计算机系统的政府治理大数据处理需要考虑混合式计算框架以及高性能计算机系统自身在体系结构、互联网络等方面的特性，因此还需要进行有针对性的研究设计。

5 研究进展及分析

5.1 政府治理大数据的管理与存储技术

大规模数据的高效管理和有效融合是实现政府治理大数据的基础设施和核心功能之一，对上层各类分析应用的数据处理能力、性能、准确度等具有重要影响。其中，管理涵盖了大规模政府治理数据集的采集和共享技术，融合涵盖了多源异构数据的质量保证和知识图谱构建技术。下面主要从数据获取、数据共享、数据质量3个角度介绍相关核心技术的研究现状。

（1）大规模多样性数据采集技术

面向政府治理的综合分析应用需要具备对多源异构异质数据的采集能力，为政府治理提供自动的数据获取手段。根据数据对象的不同，数据采集技术也有所差异，主要包括3种类型。第一种是基于时间采样的数据获取技术，负责采集位置数据、传感数据等类型的数据，焦点是采用何种感知技术准确地获取目标数据以及如何设置合理的数据采集间隔以保证采集数据能反映目标真实状态。RADAR系统[9]提供了一种基于多个基站在重叠区域内的信号强度定位室内用户的方法，进而实现室内用户跟踪。第二种是以数据爬取和数据抽取协同工作为代表的数据获取技术，主要对象是Web数据，由于Web数据的嵌入页面特征，这类数据获取技术的主要目标是有效地将目标数据从Web页面中分离并净化。SmarkCrawler[10]可从深层Web中发现并获取结构化数据；参考文献[11]提出一种从深层Web中爬取主题相关数据的方法；参考文献[12]则通过开采Web页面的可视特征提出一种新颖的数据抽取方法。第三种是基于抽取、转换和装载协同工作

的多源异构的结构化数据集成技术，目前流行的Informatics、Kettle等工具均是这种技术的代表。上述获取技术多以单一类型的数据为工作对象，面向政府治理的大数据采集涵盖政务数据、轨迹数据、Web数据等多类数据，实时性分析也对数据获取性能提出高要求，因此需要在多目标数据协同获取及其性能优化方面开展深入的研究。

（2）高并发数据共享技术

大规模数据的高并发共享具有2个研究视角：一是基于索引结构优化单次访问性能，从而整体提升数据的并发共享度；二是基于事务管理技术，通过并发控制协议以及事务特性的等级约束设置等实现高并发共享。参考文献[13-14]分别基于多核计算架构、分布式内存数据库对流行的并发控制协议进行评测，指出现有协议无法发挥多核和分布式内存的性能，需要进行优化或重新设计。Nitro[15]和STI-BT[16]均在键值（key-value）分布式数据库上通过构建索引提升读写并发性能，Nitro更充分开发了多核和大内存带来的性能优势，支持索引支持下的读写操作的线性扩展。由于面向政府治理的大数据管理平台的核心职责是向上层应用提供数据，即读操作是核心操作，因此从建立有效的分布式索引、同步优化单次操作性能和整体性能角度展开研究将是一个好的突破口。

（3）数据质量保证技术

将大量“数据孤岛”中的结构化数据进行集成与融合的最大挑战是数据一致性等质量保证问题。参考文献[17]认为数据质量保证由错误侦测和错误修复2个阶段构成，其中错误侦测技术主要有以统计方法和异常发现为主的定量分析、以模式和规则为代表的定性分析2个流派。参考文献[18]对流行的基于定量分析策略的数据质量保证方法进行了综述。在定性分析方面，参考文献[19-21]均是通过建立条件函数依赖并辅以上下文规则来净化数据的，参考文献[22]通过将函数依赖引入分布式环境实现错误侦测，具有一定的借鉴作用。而面向政府治理的大数据质量保证比一般化的大数据质量保证更有难度，首先，政务大数据的大规模、多样性使得数据质量标准本身就是一个需要研究的问题；其次，定量的政务大数据分析的计算复杂度大，而定性分析策略可能导致规则膨胀以及规则不确定性的问题。因此需要研究如何充分利用数据依赖语义、具有条件概率的数据依赖，以及数据本身的多样性等特性来设计新的数据质量标准和数据质量保证策略。

5.2 政府治理大数据分析处理技术

根据笔者的调研，目前还没有针对政府治理应用的大数据分析处理框架。现有的政府治理大数据应用基本上是基于具体的数据分析处理算法进行专门设计来实现的。

MapReduce及其衍生框架Spark、Storm是当前主流的大数据分布并行处理框架。MapReduce由Google Lab开发，能够通过分而治之的策略将不具有计算依赖关系的大数据和任务进行分割，实现并行处理。Spark和Storm则分别是面向内存计算、实时计算环境设计的。MapReduce及其衍生框架是面向分布式集群系统设计的编程模型，并行化完全依赖于并行技术，无法处理复杂的并行性应用。

而传统的超级计算框架，面向复杂的并行应用，主要采用MPI编程模型。计算框架由一个或多个彼此通过库函数进行消息收发通信的进程组成。其应用程序的并行化由程序员通过专门设计实现。但是MPI并行框架在易用性、扩展性、容错性等方面难以满足大数据处理的需求。

目前在分布并行计算框架和模型方面的一个新趋势是高性能计算机系统模式和

MapReduce模式的融合，所采取的方法主要有如下2类。

一是在超级计算机上优化MapReduce编程模型。例如，Wang等人[23]基于大数据应用使用的键值数目、维度等特征，提出一种面向多核体系结构的MapReduce库，将中间的key/value进行组合优化，实现map/reduce的多核系统优化。Micheal等人[24]实现了一个框架HPCHadoop，使Hadoop应用可以在Cray X超级计算机系统上运行。Panda等人利用超级计算机的互联通信协议加速map/reduce的通信，基于超级计算机最常用的RAMA互联实现了HiBD（high-performance big data）软件包，主要优化基于RDMA的数据shuffle、非阻塞和基于块的数据传输、Off-JVM-heap的buffer管理等。Wang等人[25]实现了基于CPU-MIC异构体系结构的MapReduce框架micMR，在向量化、内存管理、异构流水的reduce操作等方面进行了优化，体现了MapReduce在异构体系结构上的性能。

二是采用混合编程模式有效支撑应用。例如，Sandia实验室提供了一个MapReduce-MPI库，可以将一大类生物序列应用移植到超级计算机上，它为基于MPI的超算系统提供了一个开源的MapReduce的实现。有学者基于MPI实现了MapReduce的运行时系统[26-27]，将重分配和reduce过程融合，这种方法在map过程输出的键值数目有限的情况下，效果显著。

（1）机器学习算法及工具软件方面的研究

为了方便应用设计开发，已经有不少机器学习的工具软件被发布出来，主要有Caffe、Torch、Theano、TensorFlow、CNTK、MXnet、BigDL等。

Caffe是一种支持大部分机器学习算法的计算框架，底层数值计算通过高效的OpenMP/SSE/CUDA加速，同时具备灵活性和速度优势，不仅支持在CPU/GPU上运行，甚至支持嵌入式设备，如IOS、Android、FGPA。Caffe有很多衍生项目，特别是在高性能平台上的并行实现（如浪潮公司开发的Caffe-MPI、弗吉尼亚理工大学的MPI-Caffe），结合了深度学习框架以及MPI标准，使得跨越多台机器训练的深度网络变得更加简单。

TensorFlow是谷歌公司推出的第二代人工智能学习系统，它是一个利用数据流图进行数值计算的开源软件库，综合灵活，移植性好；支持Python和C++，允许在CPU和GPU上进行分布并行计算，同时支持使用gRPC进行水平扩展。BigDL是英特尔公司基于Apache Spark的开源的分布式深度学习框架，它借助现有的Spark集群运行深度学习计算，并简化存储在Hadoop中的大数据集的数据加载。TensorFlow能够利用现有的Hadoop/Spark集群运行深度学习程序，其代码可以共享到不同的应用场景中。

为了提高数据分析处理的效率，在机器学习算法、图计算算法的并行化方面有不少的研究工作。在机器学习优化方面，主要关注与深度学习相关的工作。

目前机器学习主要采用如下3类并行化方法。

第一类为数据并行，即对训练集进行划分，每个节点仅对部分数据集进行训练，最后再将所有的结果整合[28]。逻辑回归、支持向量机等算法适用于这种并行训练模式，而稀疏自动编码器、限制玻尔兹曼机（RBM）等算法因为具有内在有序性，每一次梯度更新都与前面的结果有关，所以不适用这种方法。

第二类优化方法是对学习速率采用自适应策略[29]，这种用不断改变的学习速率代替常量的做法可以减少收敛需要的迭代次数[30]。在深度学习中，随机梯度下降（SGD）算法是一种主要的最小化代价

函数算法，但是它对每一个训练样本都执行一次更新，为了克服这种样本有序性以及需要手动调整学习速率的缺点，批量方法被提出来，如限制变尺度（BFGS）算法以及共轭梯度（conjugate gradient）算法[31]，虽然更新一次参数的计算量比SGD大，但是这2种算法都提高了并行化程度。Le等人[32]在2011年对L-BFGS算法以及结合了线性搜索的共轭梯度算法进行了实验，测试了在不同硬件环境中（例如GPU或者计算集群等）2种算法的效果，实验表明卷积神经网络（convolutional neural network，CNN）在手写数字识别的训练集上的精确度有显著的提高。

第三类方法是采用异构架构，借助协处理器实现加速。自从2009年Ng A Y等人[28]首次运用GPU对无监督学习中的深度信念网络（deep belief network，DBN）以及稀疏编码（sparse coding）2个模型进行加速后，当前学术界和开源社区几乎都采用GPU并行计算平台。从2007年开始，通用图形处理器（general-purpose computing on graphics processing units，GPGPU）的普及使得众核协处理器（many-core coprocessor）成为并行处理的一个发展趋势[33]。由于众核协处理器具有强大的并行处理能力，因此采用CPU+GPU或者CPU+MIC的异构架构，让CPU负责复杂的逻辑计算部分，让GPU或MIC执行并行度高、分支少的密集运算，在学术界和工业界掀起了热潮。2014年，Jin等人[34]首次提出将Intel Xeon Phi运用于大规模深度神经网络的训练，实验结果表明Intel Xeon Phi能够提供比GPU以及Intel Xeon CPU更好的并行化效果；Andre Viebke[35]也利用Intel Xeon Phi设计了名为CHAOS的并行框架探究处理器的线程并行以及SIMD并行粒度，与GPU相比，该框架采用HogWild方法将梯度累积存储在本地，利用worker更新全局的权重参数，因此不需要明确的同步，以此充分减少卷积神经网络每一轮的训练时间，从而达到加速的目的。除了利用协处理器，还有一些利用其他硬件加速器的例子，Xia等人[36]在2016年提出一种利用阻变存储器（resistive random access memory，RRAM）以及RRAM crossbar训练卷积神经网络的方法，利用RRAM的电学特性，将CNN中层与层之间大量的中间结果量化为1 bit，并作为一个输入信号节省空间及能源；同时Bojnordi等人[37]也利用RRAM减少内存单元和计算单元的数据交换，实现深度学习中玻尔兹曼机的组合优化。

（2）图计算关键算法并行化方面的研究

在大数据分析处理过程中，与图相关的数据处理是一个重要部分。在分布并行环境下，如何对图计算的关键算法进行优化是图计算的主要研究内容。

宽度优先搜索（breadth first search，BFS）算法是图计算中最重要的算法，也是图计算系统评测标准Graph500的核心算法[38]。BFS算法的并行优化的基本方法包括减小算法访存开销、利用多线程并行搜索、隐藏通信开销3种。

Pichiorri等人[39]用位图的数据结构表示算法中的visit结构，增大了visit的局部性，减少了访存的次数。Beamer等人[40]开创性地提出bottom-up的搜索方式，避免了多线程执行的原子操作，并通过结合top-down与bottom-up进一步减少搜索过程中遍历的边数，减小了访存开销。Yasui等人[41-42]提出了内存绑定和线程绑定的优化技术，并对任务进行划分，使得多线程并行执行时各线程在搜索时尽量减少对远程的内存访问，以减小访存开销。

对于多节点的BFS算法优化，常用的方法是减少和隐藏通信开销。Yoo等人[43]在IBM BlueGene/L上实现了包含32 768个节点的分布式BFS架构，并通过边分割取

代传统的点分割，降低通信开销。Mizell等人[44]实现了128个处理器、256个处理器和512个处理器的可扩展多线程并行BFS算法，并利用硬件多线程技术来隐藏访存延迟，具有很好的性能。Ueno等人[45]利用GPU的多线程技术和细粒度同步机制对BFS算法进行加速，并采用SIMD VLQ编码方法对通信数据进行压缩，进一步提高计算性能。Convey公司[46]采用通用处理器与FPGA协处理器相结合的结构，充分利用协处理器存储器的gather/scatter能力，在主机上采用自顶向下的算法，在协处理器上采用自底向上的算法，使用数以千计数量的线程遍历图，该设计获得了非常高的性能。

Fuentes等人[47]从通信的角度对Graph500进行了分析，对消息聚合进行了评测和分析，确定了导致性能损失的原因并提出均衡方案。Eisenman等人[48]对内存子系统工作负载进行了描述，并得出结论：图的不规则性导致图计算效率偏低。而对图采用非均匀的方法进行分割，会导致各部分计算量差异较大，最终影响可扩展性。

6 面向政府治理大数据的高性能计算框架

6.1 基于混合计算模式的整体框架

针对政务大数据的多源、异构、异质特征，为满足政府治理应用对数据存储、数据访问以及数据处理的多样性需求，提出政务大数据处理框架，如**图1**所示。该框架主要包括4个模块：大数据应用、作业提交/任务管理、超算框架和MapReduce框架，以及数据存储管理系统。

面向政府治理大数据的存储管理平台是政务大数据处理框架的构成要素之一，

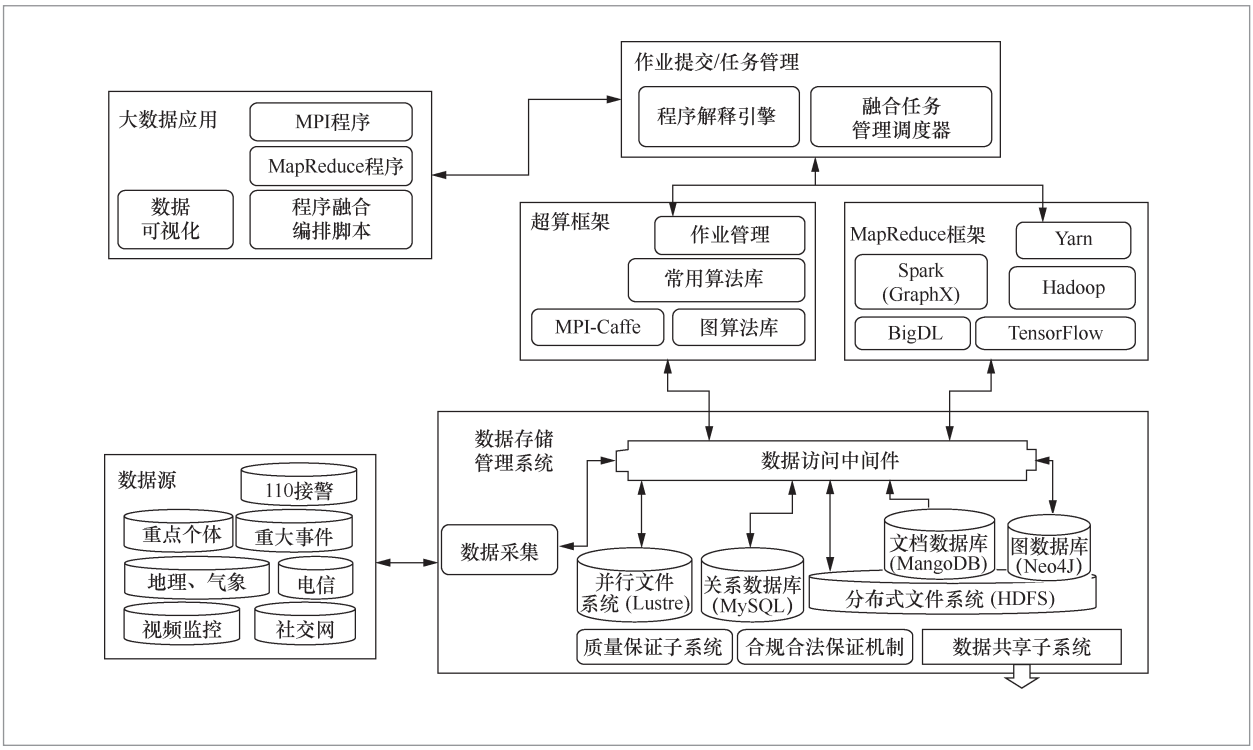

图1 政务大数据处理框架

该平台基于高性能计算机系统的计算架构特性、存储与I/O等优势，提供数据的可获得性、准确性和可用性。首先，本框架中的数据存储管理系统是混合式的大数据存储系统，能够整合封装不同的存储模型，形成统一的存储管理中间件，例如，以文件形式保存的视频监控数据，使用传统的关系型数据库保存的经济运行指标数据等。其次，不同的技术框架采用的存储方式和系统也有差别。如MPI的超算框架能将数据存储在SQL数据库和并行文件系统中，而MapReduce框架则是基于HDFS、NoSQL的。为此，上述政务大数据处理框架中的数据存储管理系统针对高性能计算机系统的存储特性，整合封装不同存储模型的存储管理中间件，实现不同存储技术、存储方式的融合。最后，针对政务大数据的特点，使用高通量、可伸缩、负载自均衡的分布式数据采集方法，以满足大规模多样性数据的实时采集需求。同时，使用多级分布式索引结构和多粒度数据共享机制，支持数据联动访问，实现政府治理大数据的高并发柔性粒度共享。

6.2 计算任务管理与运行系统

基于上面的计算框架，可以设计实现具体的计算系统。其中一个需要考虑的关键问题是如何实现计算任务的编排和管理。从现有的技术和方法来看，有如下2种不同的思路，但是均不太适用于高性能的混合大数据分析处理场景。

- 基于多种任务框架，使用脚本进行任务的生命周期和资源管理。这种方法简单快捷，适合小型和小规模任务。但是随着任务规模扩大，任务编排的业务逻辑会越来越复杂，使用脚本难以维护和调试。
- 使用统一的底层资源管理框架（如Mesos和Yarn），在其之上可以迁移和安装不同的应用框架（如Hadoop、Spark）。这样做的好处是可以由底层资源框架集中全局的资源信息，提供统一的任务和资源管理策略，管理的效率和效果都可以达到比较好的水平。但是该方案需要应用框架兼容同一个底层资源框架。以Mesos为例，目前兼容的应用框架非常有限，而且新的应用框架层出不穷，要兼容统一的底层资源管理框架需要较大的工作量，比较困难。

针对以上方法的不足，考虑高性能计算机系统在架构、网络等方面的独特性，对处理框架、处理算法进行优化设计，笔者提出一个新的混合计算模式的任务管理与运行系统MixOperator。

MixOperator用于对异构多集群计算任务进行编排管理，即提供不同类型的任务管理模式，将不同运行环境资源的任务混合编排在一起。一个依赖多种计算环境和资源的综合任务可以通过MixOperator编排完成。该系统主要由4个部分组成：主节点管理器、消息队列、从节点执行器、共享存储系统，如**图2**所示。依赖不同计算环境的计算子任务将由主管理组件发配到不同的任务消息队列中等待被调度，这些子任务将会被依赖的集群获取并运行，运行的输入和输出将通过多集群统一共享存储实现。

主节点管理器提供任务编排定义和调度的功能，可以将需要运行的任务定义信息抛给消息队列；然后，运行在不同资源环境的执行器组件可以监控自己感兴趣的消息队列，如果有需要自己运行的任务出现在自己监控的队列中，就执行相应的任务；最后执行器将需要输入和输出的文件都存储在一个共享存储系统中，这样就可以实现多种不同的系统环境之间的资源共享。

在混合式任务管理和调度机制中，通过全局性的重点考虑，根据任务和数据在不同阶段的特征，按需动态调度和配置I/O资源、计算资源、加速器资源、网络资

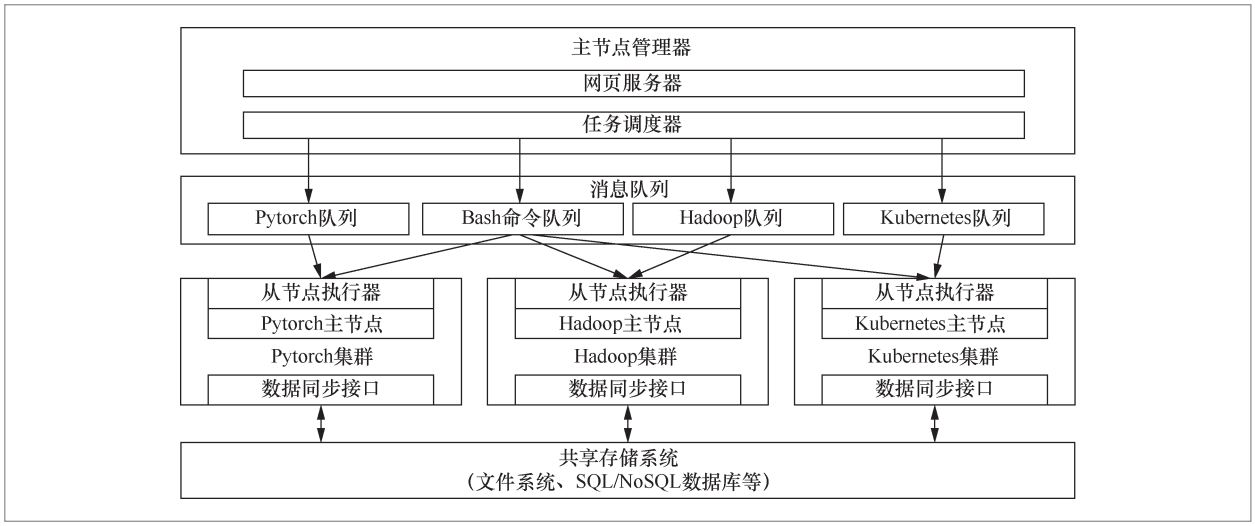

图 2 MixOperator系统的组成

源、数据与软件库资源等，以实现系统与应用的最佳匹配，高效地支撑政府治理大数据应用。特别地，需要研究MPI平台和MapReduce平台间的负载均衡调度，实现2个框架的有效统一、融合，真正发挥融合框架的优势。

MixOperator的主要优点包括：基于工作流引擎编排任务，可以用工作流规则定义任务的依赖关系和环境需求，相对脚本来说更容易维护；使用消息队列区分任务环境类型，提供松耦合、灵活的任务编排方式；针对不同的应用框架分别定制对应的存储适配器，方便将不同框架融合到统一的共享存储系统中。

7 结束语

随着技术水平的逐步提高，政府治理迈入了大数据时代。信息化技术的普及使政府拥有和管理了规模巨大的政务大数据，政府治理活动迫切需要大数据技术的支撑和保障。我国已经把大数据发展应用上升到国家战略高度。而数据的多源、异构、异质的特点以及应用场景的复杂性、多样性、多主体性，也给政府治理大数据的分析处理带来巨大挑战。利用大数据存储、分析处理等技术实现政府治理大数据的有效管理和利用，并通过相应的应用服务于政府治理需求，是政府治理大数据分析处理技术研究的主要内容。

根据政府治理场景的应用需求以及大数据技术的发展现状，政府治理大数据分析处理技术方面有待解决的关键技术问题有如下3个方面。

- 适应社会组织层次架构的政府治理大数据开放共享管理和访问。政府治理大数据的访问和共享管理需要考虑政府、企业、公民等多种类的主体及其相互之间的层次关系架构。不同的主体有不同的数据访问和处理需求，不同的主体拥有的数据也具有不同的隐私、所有权保护需求。满足这些多样复杂的需求，实现具有多样性隐私保护、多样性数据访问控制和审计的大数据共享和管理，是一个必然的趋势，也是一个巨大的挑战。
- 适应分布式多数据主体、多治理主体的政府治理大数据处理框架。在大数据

分析处理层面，政府治理应用场景的多主体问题也是一个关键难点。不同的主体拥有不同的数据，不同的主体需要不同的数据，而应用需求又要求对不同的数据进行融合处理，因此需要实现多主体数据的协同计算处理。但是，目前的研究主要集中在混合的数据处理框架方面，主要考虑的是不同的数据处理任务的计算特性，还没有考虑数据处理过程中的多主体性和多样性。

● 实现切实有效的综合性政府治理大数据分析处理系统示范应用。目前的政府治理大数据应用基本还属于针对个别政府部门、针对特定应用功能的系统，只能处理特定主体的数据，完成比较简单的目标。真正能融合多域、多主体，具有一定通用性的政府治理大数据处理技术和系统还非常少。而政府治理这样的应用领域需要通用性、基础性的应用系统，这是降低大数据技术应用的技术门槛和成本，实现大数据技术普及应用的必然要求。

参考文献:

[1] NUAIMI E, NEYADI H, MOHAMED N, et al. Applications of big data to smart cities[J]. Journal of Internet Services and Applications, 2015, 6(1): 1-15.

[2] DAHL G E, YU D, DENG L, et al. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition[J]. IEEE Transactions on Audio Speech & Language Processing, 2012, 20(1): 30-42.

[3] JUN C, CHUNG C. Big data analysis of local government 3.0: focusing on Gyeongsangbuk-do in Korea[J]. Technological Forecasting & Social Change, 2016, 110: 3-12.

[4] 夏火松, 甄化春. 大数据环境下舆情分析与决策支持研究文献综述[J]. 情报杂志, 2015(2): 1-6.
XIA H S, ZHEN H C. Public opinion analysis and decision support study under big data surroundings[J]. Journal of Intelligence, 2015(2): 1-6.

[5] JIANG D, LEUNG K, NG W. Fast topic discovery from web search streams[C]// The 23rd International Conference on World Wide Web, April 7-11, 2014, Seoul, Korea. New York: ACM Press, 2014: 949-960.

[6] 王登峰. 网络舆情事件热点发现的算法比较分析[J]. 信息通信, 2014(2): 32-34.
WANG D F. Algorithm analysis on network public opinion hotspot detection[J]. Information & Communications, 2014(2): 32-34.

[7] RATHORE M, AHMAD A, PAUL A, et al. Urban planning and building smart cities based on the Internet of things using big data analytics[J]. The International Journal of Computer and Telecommunications Networking, 2016(101): 63-80.

[8] KITCHIN R. The real-time city? big data and smart urbanism[J]. Geo Journal, 2014, 79: 1-14.

[9] BAHL P, PADMANABHAN V N. RADAR: an in-building RF-based user location and tracking system[C]// IEEE INFOCOM 2000, March 26-30, 2000, Tel Aviv, Israel. Piscataway: IEEE Press, 2000: 775-784.

[10] ZHAO F, ZHOU J, NIE C, et al. SmartCrawler: a two-stage crawler for efficiently harvesting deep-web interfaces[J]. IEEE Transactions on Services Computing, 2016, 9(4): 608-620.

[11] LIAKOS P, NTOULAS A, LABRINIDIS A, et al. Focused crawling for the hidden web[J]. World Wide Web, 2016(19): 605-636.

[12] LIU W, MENG X F, MENG W Y. ViDE: a vision-based approach for deep web data extraction[J]. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(3): 447-460.

[13] YU X, BEZERRA G, PAVLO A, et al. Staring into the abyss: an evaluation of

concurrency control with one thousand cores[J]. VLDB Endowment, 2014, 8(3): 209-220.

[14] HARDING R, AKEN D V, PAVLO A, et al. An evaluation of distributed concurrency control[J]. VLDB Endowment, 2017, 10(5): 553-564.

[15] LAKSHMAN S, MELKOTE S, LIANG J, et al. Nitro: a fast, scalable in-memory storage engine for NoSQL global secondary index[J]. VLDB Endowment, 2013, 9(13): 1413-1424.

[16] DIEGUES N, ROMANO P. STI-BT: a scalable transactional index[J]. IEEE Transactions on Parallel and Distributed Systems, 2016, 27(8): 2408-2421.

[17] CHU X, ILYAS I. Qualitative data cleaning[J]. VLDB Endowment, 2016(9): 1605-1608.

[18] HELLERSTEIN J M. Quantitative data cleaning for large databases[R]. United Nations Economic Commission for Europe (UNECE), 2008.

[19] FAN W, GEERTS F, JIA X. Semandaq: a data quality system based on conditional functional dependencies[J]. VLDB Endowment, 2008: 1460-1463.

[20] FAN W, GEERTS F, JIA X, et al. Conditional functional dependencies for capturing data inconsistencies[J]. ACM Transactions on Database Systems, 2008, 33(2): 1-48.

[21] CHIANG F, MILLER R. Discovering data quality rules[J]. VLDB Endowment, 2008(8): 1166-1177.

[22] JIN C, LALL A, XU J, et al. Distributed error estimation of functional dependency[J]. Information Sciences, 2016, 345: 156-176.

[23] QUE X, WANG Y, XU C, et al. Hierarchical merge for scalable MapReduce[C]// Proceedings of the 2012 Workshop on Management of Big Data Systems, September 21, 2012, San Jose, USA. New York: ACM Press, 2012: 1-6.

[24] MICHEAL S, THOTA A, HENSCHEL R. HPCHadoop: a framework to run Hadoop on Cray X-series supercomputers[C]// Cray User Group Meeting 2014, May 4-8, 2014, Lugano, Switzerland. [S.l.:s.n.], 2014.

[25] WANG W, WU Q, TAN Y, et al. Optimizing the MapReduce framework for CPU-MIC heterogeneous cluster[M]. Berlin: Springer International Publishing, 2015.

[26] HOEFLER T, LUMSDAINE A, DONGARRA J. Towards efficient MapReduce using MPI[C]// The 16th European PVM/MPI Users' Group Meeting, September 7-10, 2009, Espoo, Finland. Berlin: Springer-Verlag, 2009: 240-249.

[27] MOHAMED H, MARCHAND-MAILLET S. Distributed media indexing based on MPI and MapReduce[J]. Multimedia Tools and Applications, 2014, 69(2): 513-537.

[28] RAINA R, MADHAVAN A, NG A Y. Large-scale deep unsupervised learning using graphics processors[C]// The 26th Annual International Conference on Machine Learning, June 14-18, 2009, Montreal, Canada. New York: ACM Press, 2009: 873-880.

[29] SHALEV-SHWARTZ S, SINGER Y, SREBRO N, et al. Pegasos: primal estimated sub-gradient solver for SVM[J]. Mathematical Programming, 2011, 127(1): 3-30.

[30] HAZAN E, RAKHLIN A, BARTLETT P L. Adaptive online gradient descent[C]// The 20th International Conference on Neural Information Processing Systems, December 3-6, 2007, Vancouver, Canada. New York: ACM Press, 2007: 65-72.

[31] LIU D C, NOCEDAL J. On the limited memory BFGS method for large scale optimization[J]. Mathematical Programming, 1989, 45(3): 503-528.

[32] LE Q V, NGIAM J, COATES A, et al. On optimization methods for deep learning[C]// The 28th International Conference on Machine Learning, June 28 - July 2, 2011, Bellevue, USA. [S.l.:s.n.], 2011.

[33] OWENS J D, HOUSTON M, LUEBKE D, et al. GPU computing[J]. Proceedings of

the IEEE, 2008, 96(5): 879-899.

[34] JIN L, WANG Z, GU R, et al. Training large scale deep neural networks on the Intel Xeon Phi many-core coprocessor[C]// 2014 IEEE International Parallel & Distributed Processing Symposium Workshops, May 19-23, 2014, Phoenix, USA. Piscataway: IEEE Press, 2014: 1622-1630.

[35] VIEBKE A, PLLANA S. The potential of the Intel (R) Xeon Phi for supervised deep learning[C]// 2015 IEEE 17th International Conference on High Performance Computing and Communications, August 24-26, 2015, New York, USA. Piscataway: IEEE Press, 2015: 758-765.

[36] XIA L, TANG T, HUANGFU W, et al. Switched by input: power efficient structure for RRAM-based convolutional neural network[C]// 2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC), June 5-9, 2016, Austin, USA. Piscataway: IEEE Press, 2016: 1-6.

[37] BOJNORDI M N, IPEK E. Memristive Boltzmann machine: a hardware accelerator for combinatorial optimization and deep learning[C]// 2016 IEEE International Symposium on High Performance Computer Architecture(HPCA), March 12-16, 2016, Barcelona, Spain. Piscataway: IEEE Press, 2016: 1-13.

[38] YE S, TANG Y H, LIU H Z, et al. Research on algorithm optimization of Graph500 benchmark program[C]// The 19th Annual Conference on Computer Engineering and Technology and the 5th Forum on Microprocessor Technology, August 6, 2015, Harbin, China. Hunan: Hunan Science & Technology Press, 2015: 64-71.

[39] PICHIORRI F, SUH S S, ROCCI A, et al. Scalable graph exploration on multicore processors[J]. International Communications in Heat & Mass Transfer, 2010, 39(7): 937-944.

[40] BEAMER S, ASANOVIC K, PATTERSON D A. Searching for a parent instead of fighting over children: a fast breadth-first search implementation for graph500[D]. Berkeley: University of California, 2011.

[41] YASUI Y, FUJISAWA K, GOTO K. NUMA-optimized parallel breadth-first search on multicore single-node system[C]// 2013 IEEE International Conference on Big Data, October 6-9, 2013, Silicon Valley, USA. Piscataway: IEEE Press, 2013: 394-402.

[42] YASUI Y, FUJISAWA K, SATO Y. Fast and energy-efficient breadth-first search on a single NUMA system[M]. Berlin: Springer, 2014.

[43] YOO A, CHOW E, HENDERSON K, et al. A scalable distributed parallel breadth-first search algorithm on BlueGene/L[C]// The 2005 ACM/IEEE Conference on Supercomputing, November 12-18, 2005, Seattle, USA. Piscataway: IEEE Press, 2005: 25-25.

[44] MIZELL D, MASCHHOFF K. Early experiences with large-scale Cray XMT systems[C]// 2009 IEEE International Symposium on Parallel & Distributed Processing, May 23-29, 2009, Rome, Italy. Piscataway: IEEE Press, 2009: 1-9.

[45] UENO K, SUZUMURA T. Parallel distributed breadth first search on GPU[C]// The 20th Annual International Conference on High Performance Computing, December 10-21, 2013, Bangalore, India. Piscataway: IEEE Press, 2013: 314-323.

[46] WADLEIGH K, AMELIO J, COLLINS K, et al. Abstract: hybrid breadth first search implementation for hybrid-core computers[C]// 2012 SC Companion: High Performance Computing, Networking, Storage and Analysis, November 10-16, 2012, Salt Lake City, USA. Piscataway: IEEE Press, 2012: 1354.

[47] FUENTES P, BOSQUE J L, BEIVIDE R, et al. Characterizing the communication demands of the graph500 benchmark on a commodity cluster[C]// 2014 IEEE/

ACM International Symposium on Big Data Computing, December 8-11, 2014, London, UK. Piscataway: IEEE Press, 2014: 83-89.

[48] EISENMAN A, CHERKASOVA L, MAGALHAES G, et al. Parallel graph processing: prejudice and state of the art[C]//The 7th ACM/SPEC on International Conference on Performance Engineering, March 12-16, 2016, Delft, The Netherlands. New York: ACM Press, 2016: 85-90.

作者简介

吴维刚(1976-),男,博士,中山大学数据科学与计算机学院教授、博士生导师,广东省医疗大数据工程技术研究中心副主任、广州市超算与大数据重点实验室副主任,主要研究方向为云计算与边缘计算、大数据与深度学习、分布式共识与区块链等。

常亮(1980-),男,博士,桂林电子科技大学计算机与信息安全学院院长,中国计算机学会高级会员,主要研究方向为数据与知识工程、形式化方法、智能系统。

任江涛(1975-),男,博士,中山大学数据科学与计算机学院副教授,中国计算机学会会员,主要研究方向为数据挖掘、机器学习与自然语言处理。

古天龙(1964-),男,博士,桂林电子科技大学计算机与信息安全学院教授、博士生导师,国家百千万人才工程人选,教育部高等学校计算机类专业教学指导委员会副主任委员,中国人工智能学会离散智能计算专业委员会主任委员、人工智能教育工作委员会副主任委员,卫星导航定位与位置服务国家地方联合工程研究中心主任,主要研究方向为知识工程与大数据、人工智能伦理、形式化方法等。

收稿日期: 2020-02-01
基金项目: 国家自然科学基金资助项目(No.U1711263)
Foundation Item: The National Natural Science Foundation of China(No.U1711263)