面向大数据处理应用的广域存算协同调度系统

张晨浩[1,2]，肖利民[1,2]，秦广军[3]，宋尧[1,2]，蒋世轩[1,2]，王继业[4]
1. 软件开发环境国家重点实验室，北京 100191；2. 北京航空航天大学计算机学院，北京 100191；
3. 北京联合大学智慧城市学院，北京 100101；4. 国家电网有限公司大数据中心，北京 100031

摘要
以我国研发的高性能计算虚拟数据空间系统为基础，针对大数据处理应用如何统筹利用广域存储和计算资源的问题，设计并实现了一套面向大数据处理应用的广域存算协同调度系统。该系统可依据应用的计算特征和数据布局，通过存算协同、负载均衡、数据局部性感知等策略，在广域环境中协同调度应用数据和计算任务，统筹利用广域计算和存储资源，有效提升大数据处理应用的运行性能。在国家高性能计算环境中实际测试的结果表明，提出的调度方法可有效地支撑大数据处理应用，跨域目标协同识别、分子对接等典型应用的运行效率可提升3~4倍。

关键词
广域存算协同调度；大数据处理应用；虚拟数据空间；高性能计算环境

中图分类号：TP316 文献标识码：A doi: 10.11959/j.issn.2096-0271.2021050

A wide-area collaborative scheduling system oriented to big data processing applications

ZHANG Chenhao[1,2], XIAO Limin[1,2], QIN Guangjun[3], SONG Yao[1,2], JIANG Shixuan[1,2], WANG Jiye[4]
1. State Key Laboratory of Software Development Environment, Beijing 100191, China
2. School of Computer Science and Engineering, Beihang University, Beijing 100191, China
3. Smart City College, Beijing Union University, Beijing 100101, China
4. Big Data Center, State Grid Corporation of China, Beijing 100031, China

Abstract
Based on the high-performance computing global virtual data space system, a wide-area collaborative scheduling system for big data processing applications was designed and implemented. This system can address the issue of how big data processing applications unified use wide-area storage and computing resources. And it can collaborative schedule of application data and computing tasks based on the computing characteristics of the application and data layout through collaborative scheduling, load balancing scheduling, data locality scheduling strategies. By unified scheduling of application data and computing tasks in the wide-area environment, it can coordinate the utilization of wide-area

computing and storage resources, and effectively improve the running performance of big data processing applications. The actual test results in the national high-performance computing environment show that the scheduling method proposed can support big data processing applications effectively, and the running efficiency of typical applications such as wide-area target collaborative recognition and molecular docking can be increased by 3~4 times.

Key words
wide-area collaborative scheduling, big data processing application, global virtual data space, high-performance computing environment

1 引言

传统高性能计算应用（如高能物理、气象预报、生物信息等）的计算和数据量大且跨域分布，而且随着移动互联网、物联网等新一代信息技术的蓬勃发展，新兴应用（如智慧城市、精准医疗等）也不断产生大量数据且这些数据分布更加广泛，从GB级、TB级发展到ZB级，甚至YB级。这促使数据密集型和计算密集型任务的数据规模和计算规模逐步增加[1-2]，多中心协同处理海量数据正在成为发展趋势。高性能计算（high performance computing，HPC）平台也从传统的高性能计算领域逐步拓展到大数据处理领域，可有效满足大数据采集、过滤、索引、分析、处理所需的硬性要求[3]。为了满足数据处理的更大规模需求，国内外纷纷投入大量资源建立跨多超级计算中心（以下简称超算中心）的广域高性能计算环境，旨在提供规模更大、性能更强的数据处理平台，以支撑科学发现和科技创新。

美国国家科学基金会的极限科学与工程发现环境（XSEDE）[4]项目旨在将广域分散自治的多家机构互联，并实现广域资源共享，以提供更好的科学研发环境。XSEDE[4]可以存储、管理、处理海量的科学数据，为科学家提供一站式服务。作为一个集成了多种资源的单一虚拟系统，XSEDE[4]汇聚了超算、数据分析、数据存储等资源，可支持用户共享计算资源和数据，支持任务通过高性能计算机网络快速访问和检索数据，为多个领域的科学发现提供有力支持。欧洲网格基础设施（EGI）项目[5]旨在扩展欧洲在计算、存储、数据等方面的重要联合服务能力，使用网格计算技术将全欧洲广域分布的高性能、高吞吐计算资源聚合起来，实现科学数据共享，可为海量的数据以及计算资源提供统一的访问。EGI将不同欧洲国家的超算中心连接起来，以支持多学科联合的国际研究。EGI的子项目OneData[6-7]是一个全球数据管理系统，支持从个人数据管理到数据密集型科学计算的各种用例，用户可以使用全球计算中心和存储提供商支持的全球数据存储来访问、存储、处理和发布数据。我国在国家高技术研究发展计划（863计划）的支持下，依托国产高性能计算机建立了中国国家网格（CNGRID）。CNGRID由8个主节点连接而成，形成了18万亿次的计算能力，在当时全世界的网格环境中排名第二[8]。“十一五”期间，我国进一步发展了CNGRID，其计算能力达到3 000万亿次以上，有效支持了通用科学、工业仿真和生物医学等应用，促进了科学技术的发展[9]。北京航空航天大学针对国家高性能计算环境中广域分散存储资源的聚合需求以及大型计算应用对跨域全局虚拟数据空间的实际需要，研发了一个可运行于国家高性能计算环境的广域虚拟数据空间（global virtual

data space，GVDS）软件系统，解决了长期困扰我国高性能计算环境发展的广域存储管理和访问的瓶颈问题[10-11]。

作为跨域数据处理的典型平台，EGI、XSEDE、CNGRID等广域高性能计算环境正发展成进行大规模数据存储和处理的重要基础设施，高效地利用广域高性能计算环境支撑大数据的存储管理以及高效处理仍然面临如下挑战。

- 挑战1：如何形成全局数据空间，进而支持广域分散数据的存储、管理、传输、访问的统一管理。
- 挑战2：如何实现广域环境中数据与计算任务的协同调度，以优化多中心存储与计算资源的利用，支撑海量数据的跨中心高效处理。

针对挑战1，笔者所在团队已经研发了GVDS系统，该系统可聚合广域分散存储资源形成全局数据空间，以支持数据的统一管理和高效传输。针对挑战2，本文基于GVDS系统，研究了存储与计算协同调度策略，并实现了一个存算协同调度系统，该系统综合考虑数据布局、存算资源状态、容量限制等多方面因素，可合理选择任务和数据的优化调度策略，实现在广域范围内高效的计算任务分配和数据布局，以提高环境资源利用率，提升应用计算效率。

本文的主要贡献包括以下3个方面：

- 研究并实现了一套包含存算协同调度在内的调度方法，以支撑高性能大数据的快速分析处理；
- 设计实现了一个基于虚拟数据空间的存算协同调度系统，可优化广域环境中的全局资源利用，支持海量数据跨域存储管理与高效处理；
- 系统已部署于国家高性能计算环境中广域分散的5个节点，并形成测试床，验证了分子对接、跨域目标协同识别等典型大型数据处理类应用。

2 研究现状

现代科学计算和实验已变得十分复杂，工程仿真、高能物理、气象领域、基因测序研究等产生的数据量可达数百TB[12]，生产生活中的移动互联网、社交媒体等也每天产生海量的数据。采用分布式资源为大数据处理提供所需的计算能力和存储能力逐渐成为大数据处理的重要选择，但是，如何充分利用这些海量资源、发挥存储与计算资源的综合效用仍然是一个亟待解决的问题[13]。存储与计算的协同调度是解决该问题的重要方法之一。通过感知资源分布、数据分布、计算需求等，依据数据访问特征和计算特征来优化数据布局和任务布局，可有效地提高跨域存储与计算资源的利用率以及海量数据的处理效率[10]。

参考文献[13]设计了Condor-G系统，该系统是面向网格的计算资源调度系统，采用Globus和Condor使用户能够统一管理多个域内的资源，提供作业管理、资源选择、安全性和容错等能力，并提供了管理网格资源的通用接口。针对大规模数据的访问效率低、可靠性差的问题，Kosar T等人[14]开发了Stork数据管理系统，通过感知数据使用特征来进行合理的数据布局和调度，以实现对广域环境中大规模数据的高效访问，同时利用参考文献[13]中Condor-G系统提供的管理网格计算资源通用接口，实现了对广域环境中存算资源的协同管理，提高了对数据密集型应用的计算效率。

Zhao L P等人[15]基于超图分区的技术对广域环境中的存储、计算、网络资源进行协同调度，减少了广域环境中数据的传输，并且最大限度地缩短了任务完成时间，提升了广域分布式计算环境中的数据中心整体性能。参考文献[16]使用任务窃

取技术使闲置的调度器通过从超载的调度器中调度任务来平衡负载，实现了动态负载均衡的计算目标。参考文献[17]以数据访问热度为核心因素，在满足创建数据副本的条件下进行数据副本的优化布局，进而系统通过感知数据副本的布局信息进行任务的调度，提高了数据处理效率，缩短了任务完成时间。参考文献[18]提出了一种双边匹配算法，将任务与资源进行多种属性间的匹配，之后将任务调度到匹配度高的资源，减少了调度过程中的开销。参考文献[19]则以最小化数据传输为目标进行任务的调度，以减少数据传输的开销。参考文献[20]提出了一种自适应调度算法，将任务分配给一定时间内闲置的资源，避免了关键任务的低效分配，同时通过一种消息超前发送的方法节省通信时间，并进一步提高整体性能。参考文献[21]将子任务按不同规则分为任务组，任务组或单个任务被映射到不同的节点，但缺乏对任务所需存储、计算资源的考虑，导致任务和资源的相关性低，例如将计算密集的任务映射到存储能力强的节点，或者将数据密集的任务映射到计算能力强的节点，最终造成任务排队时间过长或者节点资源利用不充分等问题。

综上所述，对广域环境中的存储与计算进行协同调度是优化资源利用并提升计算效率的有效方法，但广域高性能计算中存储资源的访问效率仍然较低，存储与计算的协同性较差，难以高效应对广域高性能计算环境中复杂多变的海量数据处理需求，海量数据的跨域高效处理需要高效的存算协同调度技术。

3 广域存算协同调度系统

本文基于GVDS研发了跨域多中心存算协同调度技术和系统，可综合利用广域环境中的存储和计算资源，支持数据与计算任务的统一调度，满足大数据的高效分析处理需求。针对不同的应用场景，在框架中实现了3种不同策略的调度方法：存算协同调度方法以优化系统中的全局资源利用、最小化任务执行时间为目标，实现了任务需求与资源能力的高效匹配，可合理地进行任务与数据的联合调度，优化全局资源利用，降低任务执行时间；基于负载均衡策略的调度方法以优化系统整体计算资源利用、缩短任务响应时间为目标进行调度，实现了较优的系统平均资源利用和任务完成时间；基于数据局部性的调度方法以最小化全局数据传输为目标进行调度，充分减少了任务执行过程中的全局数据传输开销，进而缩短了系统中的任务完成时间。基于本文研究的调度策略，以及GVDS提供的全局虚拟数据空间，进一步实现了一个存算协同调度系统，将该系统与国家高性能计算环境已有全局作业调度系统对接，形成多级调度系统，综合利用广域环境存储和计算资源，优化全局资源利用，提高计算效率，支撑大数据的高效处理。

3.1 广域虚拟数据空间系统

针对广域高性能计算环境中存储与计算的协同性差导致的应用计算效率低的问题，笔者团队前期研发了高性能计算虚拟数据空间系统GVDS[10-11]。GVDS可支持对跨域分散自治资源的统一管理，为海量数据提供高性能、高可靠性存储，为广域环境中的海量数据提供全局数据视图，可有效支撑应用以统一访问模式高效访问广域分散异构的存储资源，实现广域环境中分布数据的跨域共享和协同处理，以支撑跨多超算中心协同处理的应用运行模式。目前，GVDS已在国家高性能计算环境中部署了测试床，并集成到中国国家网格门

户网站的“聚合资源运行支撑环境”AROSE平台中，用户可通过3种方式登录GVDS，并使用网格的计算资源，如**图1**所示。

GVDS可结合网格环境提供的全局作业调度，综合利用广域环境下的存储和计算资源，为跨域多中心存算协同调度提供基础，进而优化全局资源利用，满足海量数据跨域高效处理的需求，如**图2**所示。

3.2 存算协同调度策略

为了充分发挥跨域存储和计算资源的效用，满足海量数据高效处理的需求，本文提出了一种数据与计算感知的存算协同调度策略。存算协同调度指综合考虑计算任务和与数据相关的多种因素，如存储和计算资源负载、数据布局情况、网络带宽负载等，以缩短任务完成时间为目标，制定数据和计算任务协同调度的最优策略。传统的调度算法一般从负载均衡的角度和提高数据局部性的角度考虑，本文提出的存算协同调度方法则从任务、数据和资源的关系的角度出发，以资源与任务的相关性为基础，结合任务的优先级、数据的访问热度、资源的负载情况，得出任务和数据的最佳调度策略，优化任务的执行时间，以提高广域高性能计算环境中的资源利用，解决任务与资源不匹配造成的任务执行时间长的问题。

基于存算协同的调度执行过程如**图3**所示。用户提交任务到存算协同调度系统，调度计划器通过调度决策产生任务调度计划和数据调度计划，调度计划会被发送到调度执行器；调度执行器调用底层作业管理系统和存储管理系统来执行任务调度与数据调度。关键步骤包括调度决策和执行两个阶

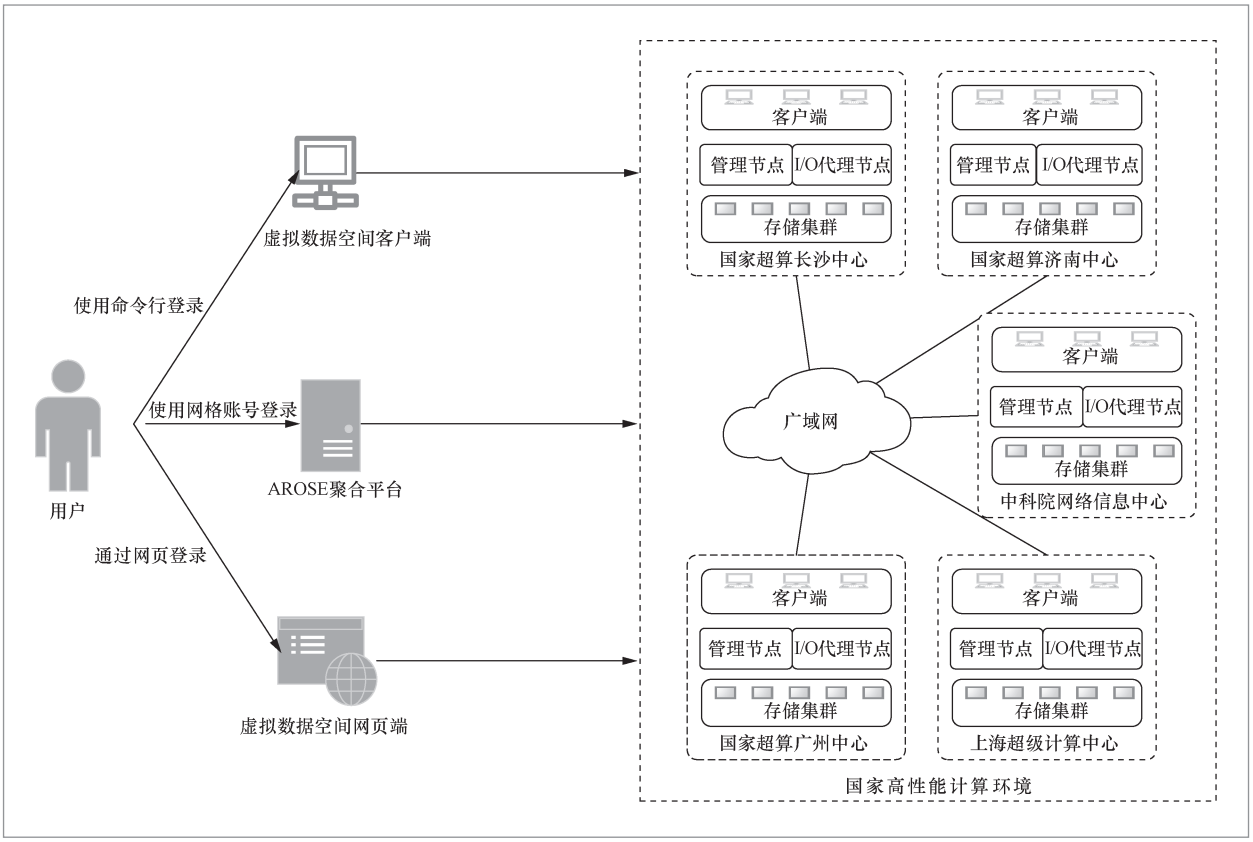


图1 GVDS与国家高性能计算环境对接

段，其中，调度决策阶段将计算任务与资源的相关性作为调度的决策依据，结合任务的优先级、数据副本布局，对任务和数据进行合理的协同调度；调度执行阶段通过Slurm的计算管理器和GVDS的存储管理器执行任务和数据的调度，同时依据数据的访问热度反馈，优化数据副本布局，以降低后续任务执行时的数据传输开销，提高计算效率。

3.2.1 调度决策阶段

（1）存算协同调度

在广域高性能计算环境中，由于任务的复杂性、存储和计算资源的多样性，系统需要合理匹配任务和资源，以产生合理的调度策略，缩短任务处理时间，提高系统吞吐量。本文提出了一种基于任务与资源相关性的协同调度方法，通过余弦相似性计算任务向量与资源能力向量的关系，选择相关性最大的任务和资源，依据节点的负载情况，协同调度任务与数据。

设$\boldsymbol{Q}$是二元向量，表示任务q所需的计算资源和存储资源，如下所示：

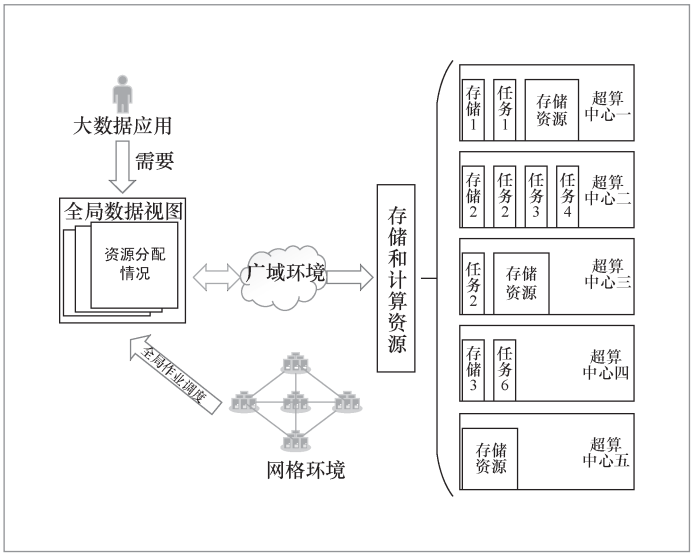


图 2 GVDS 对应用计算模式的支撑

$$\boldsymbol{Q}=(C_q,S_q) \quad (1)$$

其中，C_q是处理器核数，表示所需的计算资源；S_q是运行任务q所需的存储空间大小。

设$\boldsymbol{P}$是二元向量，表示节点p的计算能力和存储能力，如下所示：

$$\boldsymbol{P}=(\mathrm{CPU}_p,\mathrm{size}_p) \quad (2)$$

其中，CPU_p表示节点p拥有的处理器核数，

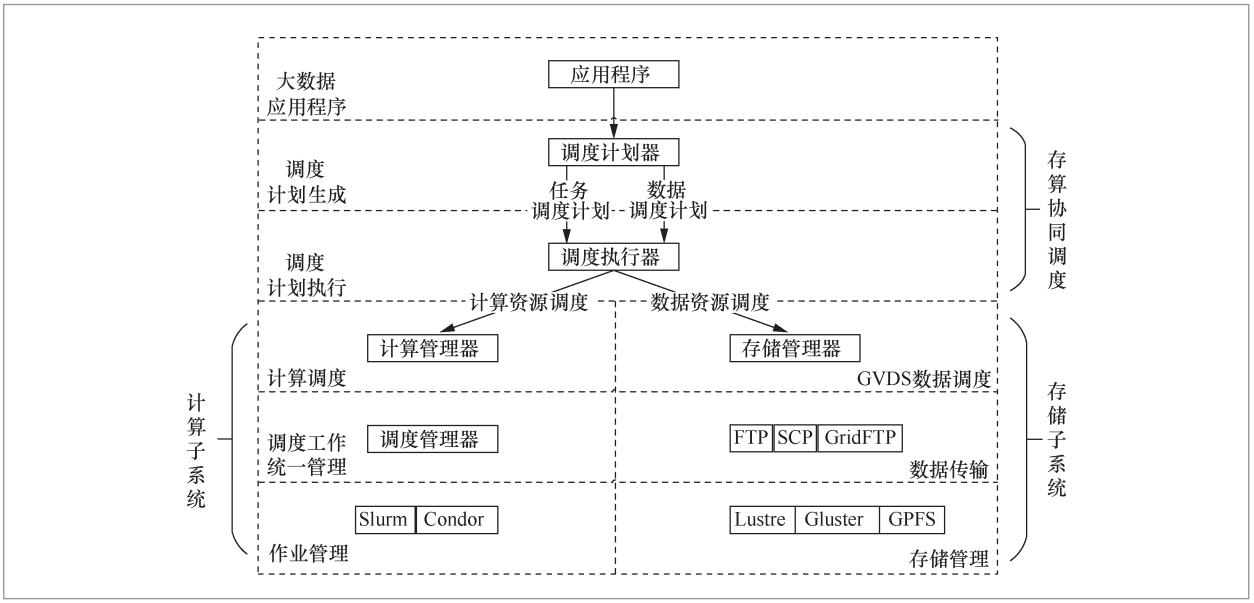


图 3 存算协同调度的执行过程

即节点p的计算能力；$size_p$表示节点p的存储空间总量。

用Rel(q,p)表示任务q和节点p的资源相关性，Rel(q,p)的表示如下：

$$\mathrm{Rel}(q,p)=\cos(\boldsymbol{Q},\boldsymbol{P}) \quad (3)$$

Rel(q,p)采用余弦函数来计算$\boldsymbol{Q}$和$\boldsymbol{P}$的相关性，值越大，任务q与节点p的资源相关性越大，即如果将任务与数据调度至该节点，预期可以缩短任务处理时间。

在实际使用场景中，如果任务所需数据距离目标节点较远，可能会出现任务等待数据传输的情况，这时候会更加倾向于优先处理可访问本地数据的任务，即数据传输距离会影响任务的优先级。因此，本文引入了任务的优先级，并将其作为任务与数据协同调度的因素。设任务q的优先级为Pri_q，如下所示：

$$\mathrm{Pri}_q=\begin{cases}1, & \text{本地数据}\\ 0.5, & \text{远程数据}\end{cases} \quad (4)$$

设置调度决策评分$Score_{q,p}$表示协同调度任务与数据的分值，如式(5)所示，采用优先级Pri_q对相关性Rel(q,p)进行加权，该式表示任务与资源相关性越大且任务的优先级越高，任务与数据被协同调度的概率就越高，从而优化资源利用并提高任务的计算效率：

$$\mathrm{Score}_{q,p}=\mathrm{Pri}_q\times\mathrm{Rel}(q,p) \quad (5)$$

(2)任务执行前的数据副本放置

在广域高性能计算环境中，受限的网络带宽导致数据传输成本很高，对于任务来说，考虑数据局部性，即将计算任务分配到数据所在的节点是合适的选择，但是对于含有大量计算任务的应用或者被频繁访问的数据来说，如果仅考虑数据局部性会导致某一节点的计算负载过高、网络负载过高、排队时间过长等，因此，在资源负载较低或者被频繁访问的节点建立数据副本是合适的选择。由Pri_q可知，数据局部性越好，任务被调度的优先级越高，因此本文的调度计划器也可基于数据访问热度，在任务执行阶段进行数据副本布局，即在任务调度阶段，依据计算资源的负载情况，预先调度数据到指定节点，以利用数据局部性，提升计算效率。

数据的访问热度由数据最近访问的时间间隔、平均访问时间间隔决定。定义平均访问时间间隔为$\overline{T_i}$，它反映数据被访问的频率；最近访问的时间间隔为(l_i-l_{i-1})，指调度策略产生时的数据访问时间l_i与上次访问该数据的时间的间隔，反映数据的访问热度趋势，(l_i-l_{i-1})越小，表明数据正逐渐成为访问热点。设数据i的访问热度为H_i，K_i表示访问数据i占所有数据的比例，则H_i的计算式如下：

$$H_i=\frac{K_i}{(l_i-l_{i-1})\times\overline{T_i}} \quad (6)$$

节点u的负载U包括节点的存储资源负载和计算资源负载，如式(7)所示。其中，U_C表示可用的计算资源比例，U_S表示可用的存储资源比例。

$$U=\sqrt[2]{U_C\times U_S} \quad (7)$$

本地节点和远程节点会根据节点的负载情况以及数据的访问热度判断是否建立数据副本。当负载情况及数据的访问热度符合设定的阈值范围时，即在节点创建数据副本；反之，则不建立。

3.2.2 调度执行阶段

(1)执行调度策略

在阶段1，调度计划器负责产生广域环境中任务需求与资源能力匹配的调度策略，调度策略被发送给调度执行器，调度执行器通过调用底层作业管理系统和存储管理系统，实际完成数据和计算任务的协同调度。计算管理器基于Slurm实现。Slurm是一个用于大型计算节点集群的高度可伸缩和容错的集群管理器和作业调度系统，提供对计算资源的监视，它将作业映射到基本的计算资源，可以实现计算任务的高效调度；存储管

理器基于GVDS实现，用于确保数据在广域范围内的统一管理、访问和传输。

（2）任务完成后的数据副本放置修正

在计算任务完成后，存算资源监控器会依据收集到的本次计算所用数据的访问热度，综合考虑节点实际的计算能力、负载情况等来修正数据副本的优化布局，以降低后续任务执行时的数据传输开销，提高计算效率。令第i份数据的平均访问时间间隔为T_i，它表示数据的访问频度，T_i越小，访问越频繁；K_i表示第i份数据被访问的次数占所有数据被访问次数的比例。第i份数据的访问热度F_i的计算式如下：

$$F_i = \frac{K_i}{T_i} \qquad (8)$$

3.3 存算协同调度系统

本文基于GVDS提供的全局统一资源管理和访问能力，以及提出的存算协同调度框架和策略，实现了一个跨域存算协同调度系统。系统的调度策略除了本文提出的存算协同调度策略外，还支持负载均衡调度、数据局部性调度，以支持高性能计算环境中的跨域任务与数据调度。

基于负载均衡策略的调度算法如**图4**所示。通过感知计算资源的全局负载进行任务调度，尽可能将计算任务均衡分配到各中心，以优化资源整体利用率，缩短任务完成时间。任务管理器将任务划分为一系列子任务，资源管理器实时检测各超算中心的计算资源负载情况，并定时将各节点的计算资源负载情况反馈到存算协同调度系统中的任务分配决策器，任务分配决策器依据各节点计算资源的空闲程度，将任务管理器划分的一系列子任务分发到不同的计算节点，以优化系统整体的计算任务分配情况。

基于数据局部性策略的调度方法如**图5**所示。任务管理器将用户提交的任务划分为一系列子任务，计算时资源管理器检测各超算中心的数据分布情况，并将各节点的数据分布情况反馈到存算协同调度系统中的任务分配决策器，通过分析系统中各子任务的类型及对数据的依赖关系，任务分配决策器将不同的子任务划分到不同的分组中，以最大化数据局部性，降低数据传输开销。

存算协同调度系统的总体架构如**图6**

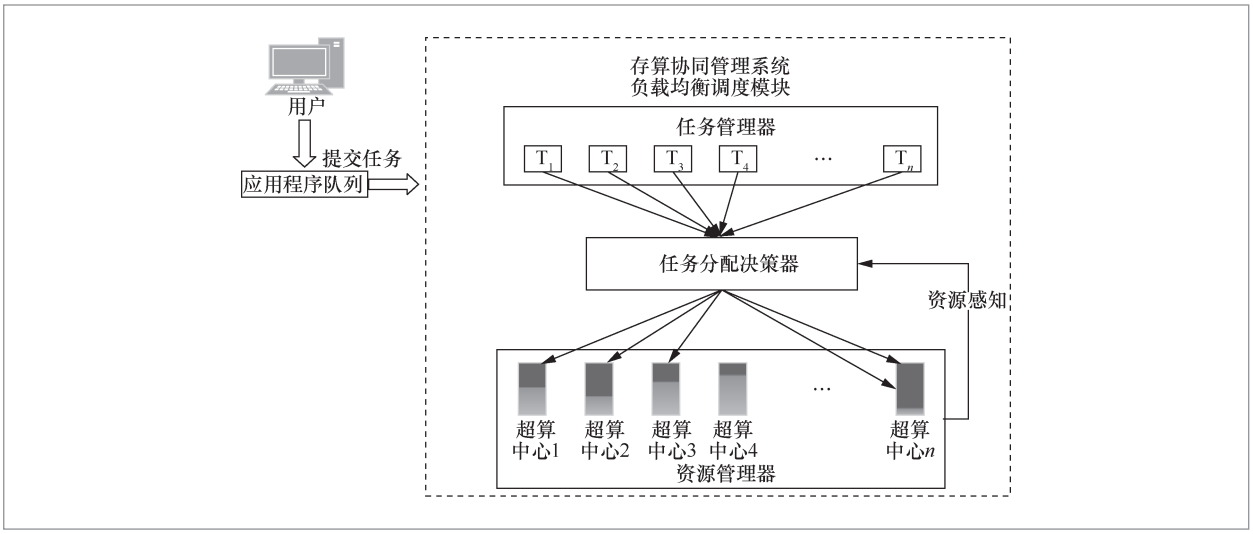


图 4 负载均衡调度策略

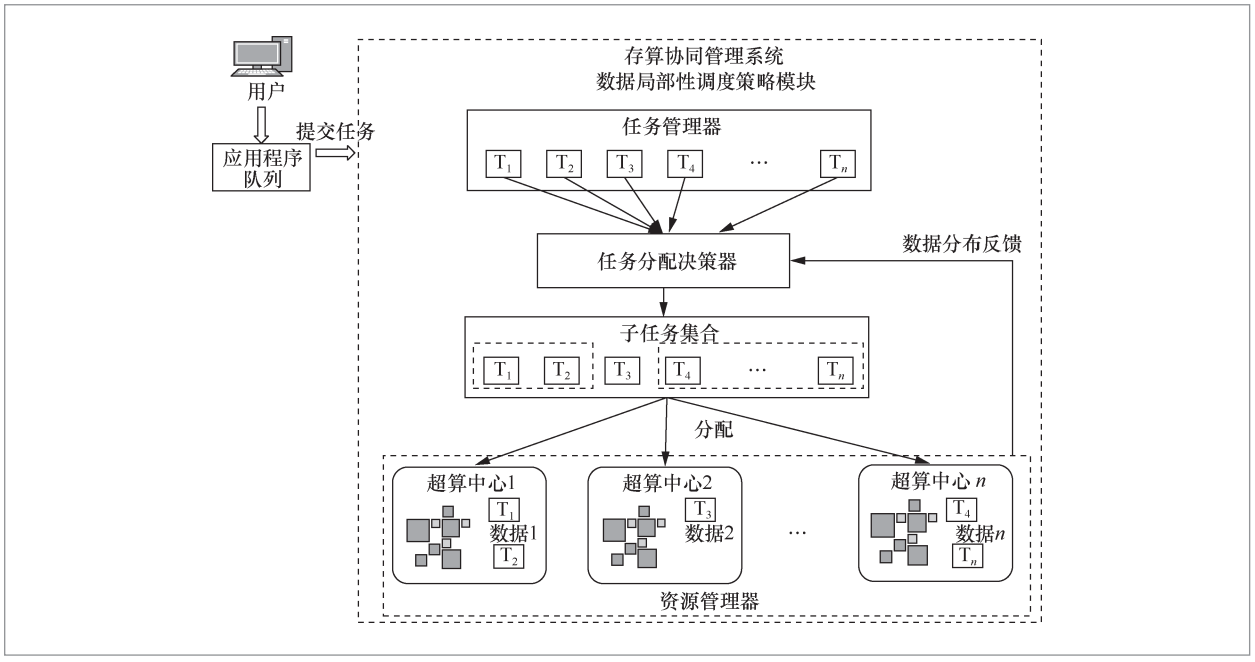


图 5 数据局部性调度策略

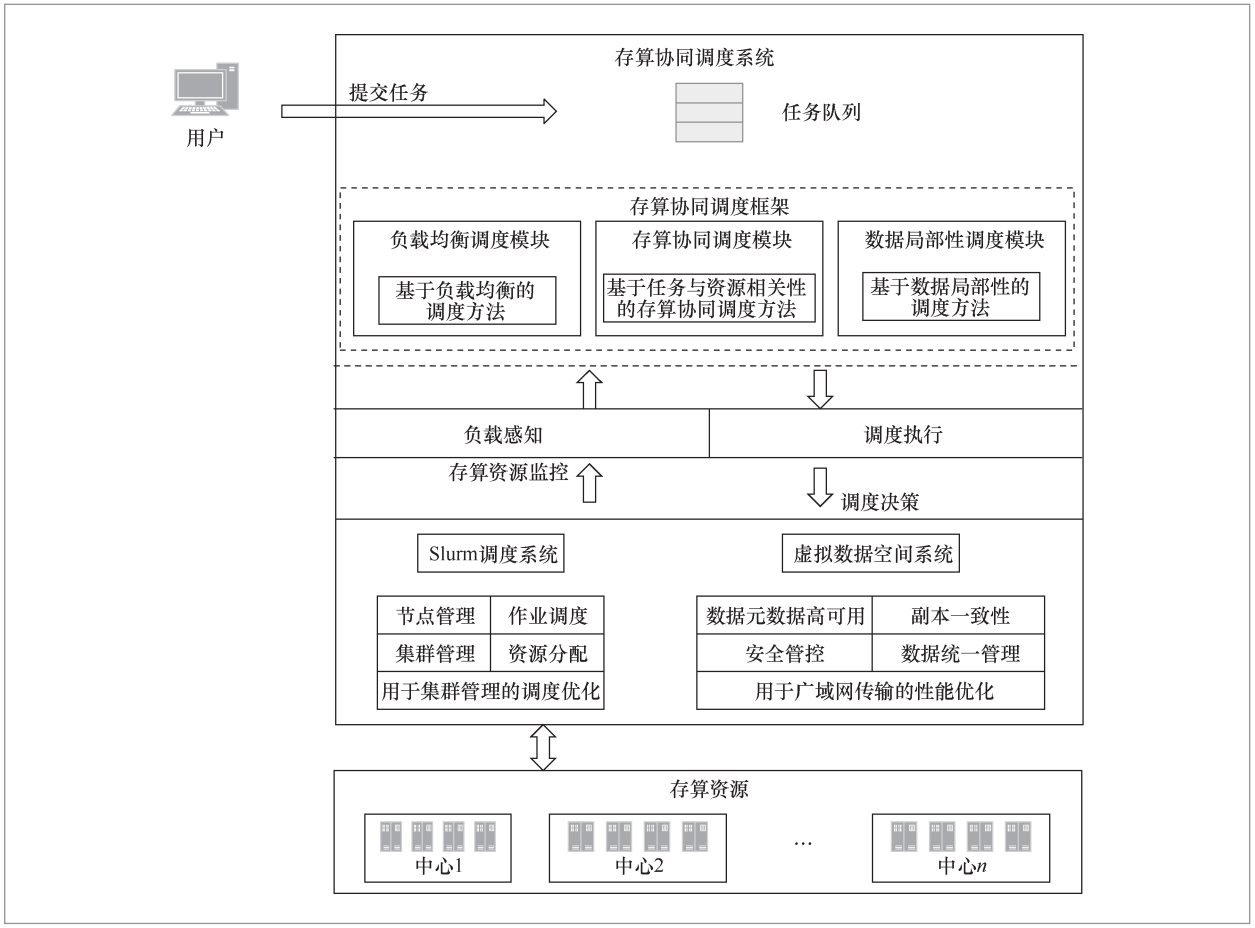


图 6 存算协同调度的系统架构

所示。该系统基于GVDS研发，目前已对接Slurm作业管理器，对负载共享设施（load sharing facility，LSF）、便携式批处理系统（portable batch system，PBS）等作业管理系统的支持也在逐步完善。存算协同调度系统对底层存算资源的状态进行监控，根据选择的调度方法产生相应的调度策略，调度策略被发布至作业管理系统和存储管理系统进行计算任务和数据的调度。

存算协同调度系统的Web界面如**图7**所示。在调度申请框中，可以选择应用的名称、任务名称、程序执行路径，CPU核心数量用于指定每个子任务占用的CPU核数，内存容量用于指定每个子任务占用的内存大小，子任务数量用于将任务划分成指定数量的子任务，调度算法可以根据不同场景选择不同的调度策略，执行参数指执行程序可选的参数，用户任务信息用于查看任务运行结果（任务名称、运行状态、开始时间、结束时间）。在任务执行状态框中，可以看到每个子任务运行所在的节点。计算资源负载框显示的是系统以轮询的方式收集的各节点的计算资源负载情况，以便系统对计算资源进行管理。

4 应用验证

目前，GVDS已在广域分布的5个国家级超算中心部署并形成测试床，本文提出的系统已在测试床上部署，并开展了分子对接应用和跨域目标协同识别应用的验证，从而证明本文提出的系统对大数据高效处理的支撑。

4.1 实验环境

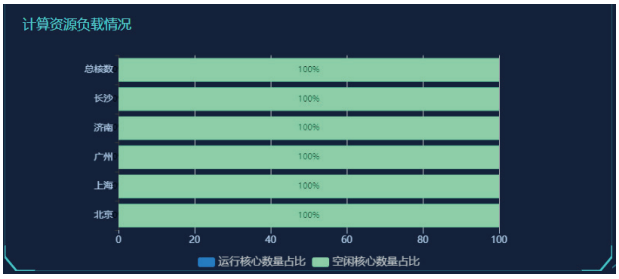
系统部署于中国科学院网络信息中心（以下简称中科院网络信息中心）、国家超级计算济南中心（以下简称国家超算济南中心）、国家超级计算广州中心（以下简称国家超算广州中心）、国家超级计算长沙中心（以下简称国家超算长沙中心）、上海超级计算中心5个国家级超算中心的测试节点，测试环境见**表1**。

4.2 应用验证

本文基于生物信息学的分子对接应

(a) 存算协同调度系统——调度申请框

(b) 存算协同调度系统——任务执行状态框



(c) 存算协同调度系统——计算资源负载框

图 7 存算协同调度系统

表1 测试环境

节点	CPU核数/个	存储容量/TB
中科院网络信息中心	112	715.75
上海超级计算中心	136	1.45
国家超算广州中心	24	1 208.121
国家超算济南中心	56	266.44
国家超算长沙中心	32	87.4

用、跨域目标协同识别应用等典型大型应用，对笔者团队研发的存算协同调度系统开展了应用验证，以研究系统对应用计算效率的提升情况。分子对接是通过受体的特征以及受体和药物分子之间的相互作用的方式来进行药物设计的方法[22]。分子对接应用的传统执行方式是在单个数据中心进行集中式的计算，这导致资源利用不均、计算效率低，而本文的存算协同调度系统可以将计算任务及数据进行合理的分配，优化资源利用率，提高应用计算效率。跨域目标协同识别应用需要对大量视频帧进行目标检测，搜寻一个目标时往往需要多个中心的数据，其计算量和数据量都较大。

首先，将分子对接应用基于存算协同调度系统提交运行，以验证系统的存算调度系统功能，实验如**图8**所示。在调度申请阶段，将分子对接应用分成100个子任务，每个子任务分配一个CPU核心，经过系统调度之后，分子对接应用的任务被分配到3个节点执行（**图8**中任务执行状态框），从**图8**中计算资源的负载情况框中可以看到各中心的负载情况。

其次，分子对接应用和跨域目标识别应用分别与两个应用的单节点运行模式相比，分子对接应用的任务被存算协同调度分配到3个节点执行，跨域目标协同识别应用被存算协同调度分配到5个节点执行，执行结果如**图9**和**图10**所示。

在测试结果中，分子对接应用基于存算协同调度系统运行的效率达到了传统运行模式的3.07倍，跨域目标协同识别应用运行的效率达到了传统运行模式的4.03倍，表明存算协同调度系统可将计算任务及数据进行合理的分配，有效地提高应用的计算效率。

最后，为了对比本文提出的存算协同调度系统提供的3种调度策略的性能，在分子对接实验中进行了单节点运行、存算协同调度策略、负载均衡调度策略、数据局部性调度策略的对比，将应用分别划分为100、200、300、400、500个子任务，分别测量3种调度方法下的任务完成时间，实验结果如**图11**所示。

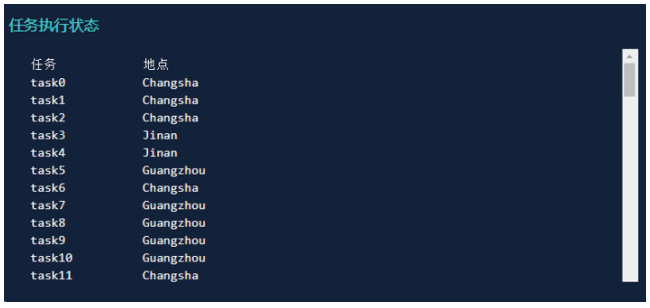
在**图11**中，当任务的计算量较小时（如**图11**中子任务数量为100~200个），各节点的计算资源状态相对空闲，任务的数据迁移时间是系统的性能瓶颈，此时，考虑了数据迁移优化的数据局部性调度策略和存算协同调度策略拥有较好的性能。这是因为存算协同调度策略在产生调度策略时，通过综合分析数据局部性和计算资源的负载情况，将任务与数据协同调度到合适的节点，在减少广域网环境中数据迁移开销的同时，避免了计算任务排队过长的情况，因此性能较优；数据局部性调度策略会将计算任务尽可能分配到数据所在的节点，避免了大量的广域网数据传输，优化了数据迁移过程，相对空闲的计算资源可以及时处理分配的计算任务，从而缩短任务完成时间，但随着计算量的增大，数据所在的节点会逐渐成为系统的瓶颈，使得任务完成时间延长，因此性能会逐渐下降；而负载均衡调度策略在计算资源相对空闲的情况下，为了均衡广域环境中各节点的负载，会产生不必要的数据迁移，出现计算任务等待数据迁移的情况，因此性能相对较差。随着任务的计算量增大（如**图11**中子任务数量为300~500个），计算资源逐渐成为系统的性能瓶颈，此时考虑了资源优化利用的存算协同调度策略和负载均衡调度策

略拥有较好的性能，存算协同调度策略拥有最好的性能，负载均衡调度策略的性能次之，数据局部性调度策略的性能较差。这是因为存算协同调度策略会基于数据局部性选择相关性最大的任务和资源，依据节点的负载情况协同调度任务与数据，优化了广域高性能计算环境中的资源利用，因此性能最优；而负载均衡调度策略会将计算任务尽可能均衡地分配到各节点，但未充分考虑任务及其所需的数据与超算中心内的计算和存储资源状态不匹配造成的任务分配不合理的问题，因此性能次之；而数据局部性调度策略在计算任务的计算量较大的情况下，会出现数据所在节点的计算任务排队过长的情况，从而导致任务完成时间大幅延长。

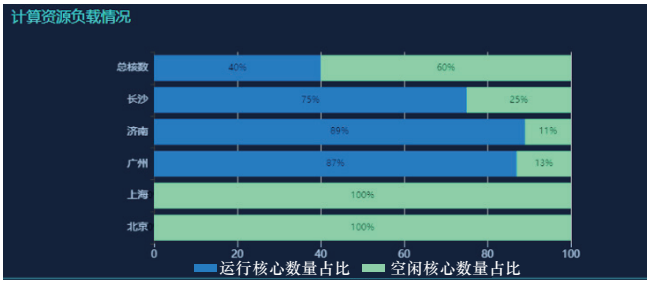
总体上，存算协同调度策略在各种子任务划分下，性能均优于其他调度方法。单节点运行的完成时间会随着任务数量的增加而大幅延长，尤其在任务数量超过300个时，由于节点负载增加，完成时间快速延长。其他3种调度策略则随着子任务数的变化而变化平缓，说明3种方法都能较好地匹配数据与计算作业。存算协同调度策略的最好情况是子任务数为500个时，完成时间仅为单节点运行完成时间的20.76%，最坏情况是子任务数为300个时，完成时间为单节点运行完成时间的52.54%；负载均衡调度策略的最好情况是子任务数为400个时，完成时间为单节点运行完成时间的25.42%，最坏情况是子任务数为100个时，完成时间为单节点运行完成时间的74.52%；数据局部性调度策略的最好情况是子任务数为400个时，完成时间为单节点运行完成时间的36.92%，最坏情况是子任务数为100个时，完成时间为单节点运行完成时间的为105.7%。

综合上述验证实验可知，存算协同调度策略可以通过合理的任务与数据调度，优化多中心存储与计算资源的利用，既避免了计算资源成为系统的瓶颈，又避免了大量数据的迁移开销，使得任务完成时间最短。

(a) 分子对接实验调度申请



(b) 分子对接实验调度结果查看



(c) 分子对接实验的计算资源负载情况

图 8 存算协同调度系统进行分子对接实验

5 结束语

本文针对海量数据的高效处理需求，

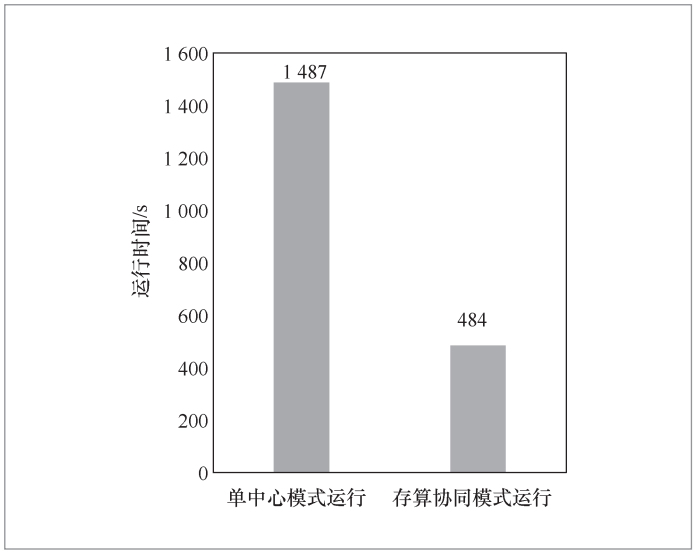


图 9 分子对接应用实验

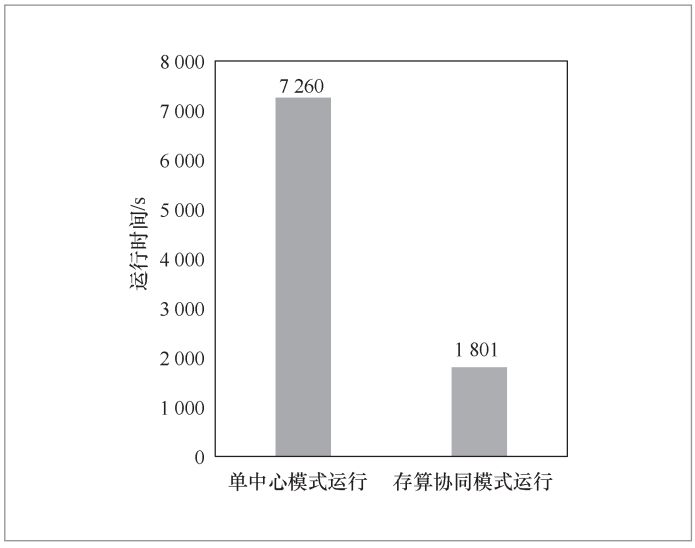


图 10 跨域目标协同识别应用实验

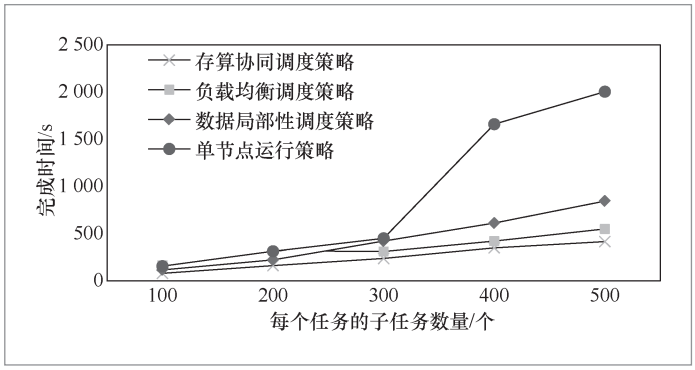


图 11 4 种调度策略性能对比

基于笔者团队研发的高性能虚拟数据空间系统，基于负载均衡调度、数据局部性调度、存算协同调度3种调度方法研发了存算协同调度系统。该系统可以统一管理广域环境中的存算资源，进行存算协同调度，优化广域环境中的存算资源利用，支持大数据的存储管理与高效分析处理。

目前，笔者团队研发的存算协同调度系统已经和GVDS一起实验性地部署于国家高性能计算环境中的中科院网络信息中心、上海超级计算中心、国家超算济南中心、国家超算长沙中心、国家超算广州中心5个超算中心的测试节点，并通过分子对接应用、跨域目标协同识别应用验证了系统的有效性和高效性，初步建成了跨域海量数据处理的实验平台。

笔者团队后续将在存算协同调度策略对高性能计算环境的资源感知方面开展工作，以进一步提高存算协同调度策略的调度精度和准确性，并集成更多的调度策略，以扩展存算协同调度系统的应用场景和灵活性，以优化国家高性能计算环境的资源利用，并为海量数据的跨域高效协同处理提供支撑。

致谢

感谢项目团队的各位老师和同学，以及为项目研发提供指导的各位项目专家，尤其感谢中山大学陈志广老师团队提供的分子对接应用、跨域目标协同识别应用。

参考文献:

[1] 佩瑟鲁·拉吉, 阿诺帕马·拉曼, 德维亚·纳加拉杰, 等. 高性能计算系统与大数据分析[M]. 齐宁, 庞建民, 张铮, 等, 译. 北京: 机械工业出版社, 2019: 17-20.

RAJ P, RAMAN A, NAGARAJ D, et al. High-performance big-data analytics computing systems and approaches[M]. Translated by QI N, PANG J M, ZHANG Z, et al. Beijing: China Machine Press, 2019: 17-20.

[2] 彭宇, 庞景月, 刘大同, 等. 大数据: 内涵、技术体系与展望[J]. 电子测量与仪器学报, 2015, 29(4): 469-482.
PENG Y, PANG J Y, LIU D T, et al. Big data: connotation, technical framework and its development[J]. Journal of Electronic Measurement and Instrumentation, 2015, 29(4): 469-482.

[3] 陈国良, 毛睿, 蔡晔. 高性能计算及其相关新兴技术[J]. 深圳大学学报(理工版), 2015, 32(1): 25-31.
CHEN G L, MAO R, CAI Y. High performance computing and related new technologies[J]. Journal of Shenzhen University Science and Engineering, 2015, 32(1): 25-31.

[4] TOWNS J, GAITHER K, BLOOD P, et al. XSEDE: extreme science and engineering discovery environment(OAC 15-48562)[R]. 2020.

[5] NEWHOUSE S. Seeking new horizons: EGI's role in 2020(EGI-1098-D230-V3)[R]. 2021.

[6] VILJOEN M, DUTKA Ł, KRYZA B, et al. Towards European open science commons: the EGI open data platform and the EGI DataHub[J]. Procedia Computer Science, 2016, 97: 148-152.

[7] WRZESZCZ M, TRZEPLA K, SŁOTA R, et al. Metadata organization and management for globalization of data access with onedata[C]//Parallel Processing and Applied Mathematics. Cham: Springer, 2016: 312-321.

[8] 历军. 高性能计算应用概览[M]. 北京: 清华大学出版社, 2018: 304-307.
LI J. Overview of high-performance computing applications[M]. Beijing: Tsinghua University Press, 2018: 304-307.

[9] XU Z W, CHI X B, XIAO N. High-performance computing environment: a review of twenty years of experiments in China[J]. National Science Review, 2016, 3(1): 36-48.

[10] 秦广军, 肖利民, 张广艳, 等. 面向国家高性能计算环境的虚拟数据空间系统[J]. 大数据, 2021, 7(2): 101-122.
QIN G J, XIAO L M, ZHANG G Y, et al. Virtual data space system for national high-performance computing environment[J]. Big Data Research, 2021, 7(2): 101-122.

[11] 肖利民, 宋尧, 秦广军, 等. GVDS: 面向广域高性能计算环境的虚拟数据空间[J]. 大数据, 2021, 7(2): 123-146.
XIAO L M, SONG Y, QIN G J, et al. GVDS: a global virtual data space for wide-area high-performance computing environments[J]. Big Data Research, 2021, 7(2): 123-146.

[12] HEY T, TREFETHEN A. The data deluge: an e-science perspective[M]// Grid computing: making the global infrastructure a reality. Chichester: John Wiley & Sons, Ltd, 2003: 809-824.

[13] FREY J, TANNENBAUM T, LIVNY M, et al. Condor-G: a computation management agent for multi-institutional grids[C]// Proceedings of the 10th IEEE International Symposium on High Performance Distributed Computing. Piscataway: IEEE, 2001: 55-63.

[14] KOSAR T, BALMAN M. A new paradigm: data-aware scheduling in grid computing[J]. Future Generation Computer Systems, 2009, 25(4): 406-413.

[15] ZHAO L P, YANG Y N, MUNIR A, et al. Optimizing geo-distributed data analytics with coordinated task scheduling and routing[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(2): 279-293.

[16] WANG K, QIAO K, SADOOGHI I, et al. Load-balanced and locality-aware scheduling for data-intensive workloads at extreme scales[J]. Concurrency and Computation: Practice and Experience, 2016, 28(1): 70-94.

[17] LI C L, BAI J P, TANG J H. Joint optimization of data placement and scheduling for

improving user experience in edge computing[J]. Journal of Parallel and Distributed Computing, 2019, 125: 93–105.
[18] HE L, QIAN Z C. Intent-based resource matching strategy in cloud[J]. Information Sciences, 2020, 538: 1–18.
[19] BRYK P, MALAWSKI M, JUVE G, et al. Storage-aware algorithms for scheduling of workflow ensembles in clouds[J]. Journal of Grid Computing, 2016, 14(2): 359–378.
[20] HU M L, LUO J, WANG Y, et al. Adaptive scheduling of task graphs with dynamic resilience[J]. IEEE Transactions on Computers, 2017, 66(1): 17–23.
[21] 尹伶艳. 广域云环境下数据与计算的协同调度[D]. 天津: 天津大学, 2014.
YIN L Y. Joint scheduling of data and computation in geo-distributed cloud systems[D]. Tianjin: Tianjin University, 2014.
[22] SAIKIA S, BORDOLOI M. Molecular docking: challenges, advances and its use in drug discovery perspective[J]. Current Drug Targets, 2019, 20(5): 501–521.

作者简介

张晨浩（1997– ），男，北京航空航天大学计算机学院博士生，主要研究方向为高性能计算、分布式存储等。

肖利民（1970– ），男，博士，北京航空航天大学计算机学院教授、博士生导师，计算机科学技术系主任，计算机系统结构研究所副所长，中国计算机学会（CCF）大数据专家委员会委员、高性能计算专业委员会常务委员、容错计算专业委员会委员，中国电子学会云计算专家委员会委员，主要研究方向为计算机体系结构、大数据存储、高性能计算等。曾获国家科技进步奖二等奖4项、省部级科技进步奖一等奖4项及其他省部级奖项5项。发表SCI/EI论文230多篇，申请发明专利100多项，其中授权发明专利88项。

秦广军（1977– ），男，博士，北京联合大学智慧城市学院讲师，CCF会员，主要研究方向为高性能计算、存储系统、大数据和机器学习等。作为项目骨干参与多项国家863计划项目、国家重点研发计划项目、国家自然科学基金面上项目、北京市自然科学基金面上项目等。

宋尧（1994– ），男，北京航空航天大学计算机学院博士生，主要研究方向为高性能计算、分布式存储、分布式调度系统、存算联动调度等。

蒋世轩（1999– ），男，北京航空航天大学计算机学院硕士生，主要研究方向为分布式存储、存算联动调度等。

王继业（1964– ），男，博士，国家电网有限公司大数据中心教授级高级工程师，主要从事电力信息化、能源互联网、大数据与人工智能等方面的研究工作。

收稿日期： 2021-05-31
通信作者： 秦广军，qingj@buaa.edu.cn
基金项目： 国家重点研发计划资助项目（No.2017YFB1010000）
Foundation Item: The National Key Research and Development Program of China (No.2017YFB1010000)