

云计算中任务调度研究的调查

马小晋 饶国宾 许华虎

(上海大学计算机工程与科学学院 上海 200444)

摘要 云计算通过虚拟技术将各类计算资源从底层硬件中剥离出来并进行动态扩展,以按需付费的方式提供给用户使用。云平台由不同的硬件架构和巨大的数据资源组成,当用户所提交的任务数量逐步增长时,如何通过调度算法对其进行有效调度并合理分配资源成为云计算中的关键环节。首先对云计算及其任务调度进行概要介绍,描述调度流程、主要算法和评测指标;随后根据不同指标和算法对近年来的相关文献进行调研概述,归纳对比了一些算法的主要特点;在此基础上,提出了未来研究所面临的几个关键环节。在实际应用中,需要根据任务和资源的不确定性和动态变化情况灵活采取调度策略,并尽可能考虑多个性能指标,综合提高云计算的运行效率和服务质量。

关键词 云计算,任务调度,虚拟化,服务质量,服务水平协议

中图分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2019.03.001

Research on Task Scheduling in Cloud Computing

MA Xiao-jin RAO Guo-bin XU Hua-hu

(School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China)

Abstract In cloud computing, virtualization technology separates various kinds of computing resources from the underlying infrastructure and expands them dynamically, and it allows users to pay on the basis of usage. Cloud platform is a heterogeneous system which consists of different hardware and huge data resources. With the increasing number of tasks, it is critical to schedule users' tasks and allocate resources effectively through task scheduling algorithm. This paper illustrated a brief introduction of cloud computing, task scheduling algorithm and the core scheduling process including evaluation metrics with some figures. Then, it proposed an overview of the related literatures and algorithms in recent years. Finally, this paper presented some key aspects of the research. In realistic applications due to the varying situation of tasks and uncertainty in resources, it is crucial to select the scheduling strategy accordingly, and taking more performance indicators into consideration can enhance the efficiency and quality of service in cloud computing.

Keywords Cloud computing, Task scheduling, Virtualization, QoS, SLA

1 引言

云计算采用虚拟化技术对网格计算、并行计算和分布式系统进行了改进和提升^[1],具有可伸缩、虚拟化和按需服务等特性,成为了满足用户需求的一种新型计算模式。在云时代,用户减少了在基础设施上的投入和维护成本,更侧重于根据实际情况按需使用云计算中的各类资源(如 CPU、存储、网络等),同时能够根据任务情况动态调整计算性能。云计算所带来的优势不仅在于降低了 IT 成本、提高了业务灵活性,更能够满足各类应用的不同需求,极大地提高了数据的多样化处理和高性能计算。

任务调度(task scheduling)是云计算中的一个关键环节,

以便各种任务经过优化后能够映射到合适的资源上执行^[2]。具体来说,任务调度是指在云计算环境中根据任务和资源的实际情况,将任务分配或迁移到相应资源上执行的过程。其中涉及了优先权、执行时间、完成时间、资源利用率、成本、能耗、网络吞吐率以及公平性等优化参数和评测指标。任务调度策略不仅直接对任务执行时间和成本产生作用,还会影响到整个云计算平台的性能。因此,如何合理分配云平台中的各种资源、高效调度用户任务,成为了云计算研究领域中的一个重要问题。近年来,学者们已提出了不少良好且具有实际应用价值的任务调度策略,但目前还没有形成一个全面、通用且有效的任务调度算法。

国内外研究人员从不同角度,利用多种方式对云环境下

到稿日期:2018-03-19 返修日期:2018-05-22 本文受赛尔网络下一代互联网技术创新项目(NGII20170513, NGII20170206),上海张江国家自主创新示范区专项发展资金重点项目(201411-ZB-B204-012)资助。

马小晋(1982-),男,博士生,主要研究方向为云计算、并行处理等,E-mail: xjma@shu.edu.cn;饶国宾(1985-),男,硕士生,主要研究方向为图像处理、云渲染等;许华虎(1966-),男,博士,教授,博士生导师,主要研究方向为人机交互、多媒体网络技术和高性能计算等,E-mail: huahuxu@163.com(通信作者)。

的任务调度问题进行了研究,提出了各种算法和技术方案。随着相关文献的日益增加和丰富,归纳和整理云计算中任务调度研究的发展状况也成为了一个重要的研究内容,这将有助于该领域的深入发展。

近几年来已有学者进行了相关论述,例如文献[3]主要讨论了云计算中资源调度的相关研究和调度方法,分别从以降低能耗为目标的调度方法、以提高资源利用率为目标的调度方法、基于经济学的资源管理模型研究和其他相关研究4个方面进行了调研论述,并对一些调度方法进行了归纳对比;但其缺乏对云计算调度自身特性的概括和分析,同时也没有对主要的调度算法和评测指标进行相应的归纳和总结。文献[4]根据云计算任务调度的相关文献对不同的调度算法逐一进行了评述,据此提出了能效和负载均衡、用户服务质量(Quality of Service, QoS)、安全性和公平性4个调度特征,并汇总概括了文献所涉及的参数指标;但所调研文献存在一定的局限性,没有再次进行归纳和分类,同时所形成的调度指标有待深入讨论和完善。文献[5]通过分析云计算资源架构,将调度划分为应用层、虚拟层和部署层3类结构,并分别分析了每一层的调度目的和研究状况,然后采用进化计算方法从调度问题和解决方案两方面对相关文献进行了综述;但其中所涉及的调度算法较少,缺乏对其他类型算法和技术方案的文献调研。文献[6]介绍了云计算的服务模型和任务调度的主要框架,将云计算调度算法划分为启发式、截止日期、优先级和优化方法4个类型,并据此对相关文献进行了调研分析,概述对比了各种调度算法的流程和特点;但未能进一步进行深入分析与总结,缺乏对未来研究方向和关键环节的论述探讨。

尽管还有一些文献对云计算任务调度的研究情况进行了调研论述,但相对于对实际算法的研究进展,此类综述性文献仍较为缺乏且滞后,已有的相关研究也存在调研不足、侧重不一等情况。据此,本文根据云计算中的主要任务调度算法对近年来的相关文献进行了归纳整理,从时间、成本、能耗等不同性能评测指标入手分别进行概述对比,探讨云计算中任务调度算法的主要特点和研究趋势。

2 云计算下的任务调度

2.1 云环境特征

云计算一般包含设施即服务(Infrastructure as a Service, IaaS)、平台即服务(Platform as a Service, PaaS)和软件即服务(Software as a Service, SaaS)3个层次的服务^[7]。这些服务通常基于按需付费使用这一原则,即用户按照云供应商所提供的不同规格,根据自身需求购买相应的计算资源,并在一定时间内使用。对用户来说,无需在基础设施上进行投入,就能够按需获得计算、存储和网络等各种资源来完成业务,而云供应商也可以最大化各类资源的利用率,增加服务效益。资源的虚拟化、分布式和动态扩展是云计算的几个主要特点^[8]。

云计算的主要结构如图1所示。用户通过客户端、网页或手机应用等特定入口访问服务,根据云平台所提供的服务内容提交任务,并按需设置价格(成本)、计算资源、截止时间

等相关参数。平台前端模块通过用户接口汇总各类任务,在满足用户服务质量的前提下,依据相关参数,按照调度策略在后台资源池中选择合适的资源(通常为虚拟机)来运行相应的任务,最终在任务完成后释放资源,并将结果返回给用户。云供应商负责提供硬件节点和配套设施来构建整个云计算平台,并在系统运行过程中通过任务调度、资源优化分配、软硬件监控和故障维护等方式保证任务和平台的稳定运行。

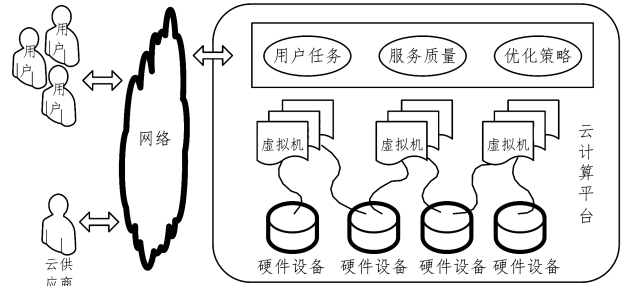


图1 云计算的主要结构

Fig. 1 Main structure of cloud computing

2.2 任务类型

云计算中的任务一般可分为独立任务和非独立任务两类,在进行任务调度时首先需要考虑任务间的相互关联性。当任务彼此之间相互独立、互不影响时,可称其为独立任务。对于此类任务,调度算法只需要依据每个任务的成本、执行时间、最后期限或优先级别等因素,采取一定策略将其分配到合适的虚拟机上运行即可,每个任务的执行结果不影响其他任务的执行。当任务之间存在一定的先后顺序或彼此影响时,可将其视为非独立任务。其特点是任务的所有前驱任务执行完成后才能开始执行该任务,通常使用有向无环图(Directed Acyclic Graph, DAG)来表示。此外,还可分为大体量、科学计算、任务组(包)等多种任务类型。

2.3 调度流程

由于云计算具有动态特性,因此当大批量用户共享云资源且同时进行任务调度时,会给云计算带来挑战^[9]。在云环境中,任务调度算法的主要目标是在用户任务和虚拟机之间找到一个最优映射,以便在满足用户需求的前提下尽可能减少所有用户任务的整体完成时间或成本,提高云资源的使用效率并降低能耗。通常情况下,任务调度要先根据成本、截止时间、执行时间等因素设置任务的优先级别,再根据优先级别依次将任务分配到满足条件的虚拟机上执行,并通过服务水平协议(Service Level Agreement, SLA)对调度质量进行评估^[10]。

在云计算中,调度体现着整个系统的效率、稳定性和灵活性等特点,因此具有十分重要的作用。任务调度算法可以以用户应用或平台资源为主进行划分。当以用户应用为主时,调度算法的主要目的是最小化成本和时间等指标;而以平台资源为主时,调度算法侧重于提高各种资源的利用率和稳定性等,并降低能耗,减少开销。图2所示为云计算平台中任务调度的一般过程。

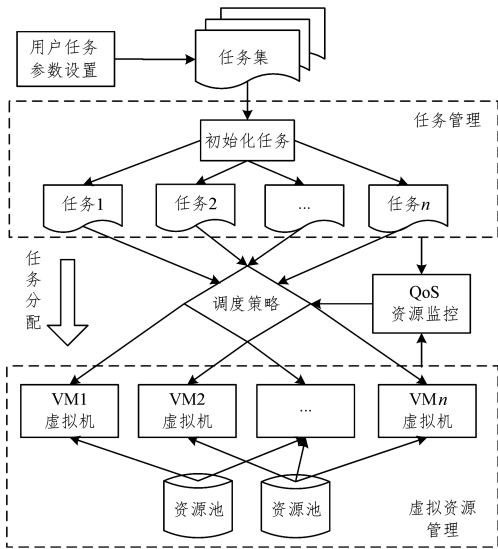


图 2 云环境下任务调度的一般过程

Fig. 2 General process of task scheduling in cloud environment

首先,用户提交的任务根据成本、时间等要素指标初始化后被加入到任务队列中,调度程序获取以虚拟机方式为主的可用资源信息;然后,在满足一定用户服务质量的前提下,根据调度策略将任务分配到合适的虚拟机上执行,通常需要综合考虑成本、时间、资源利用率和能耗等多种指标以期实现最优组合;最后,在虚拟机上运行任务,并将最终结果返回给用户。在这一过程中,调度程序还需要根据负载均衡、性能变化、任务重置、虚拟机再分配等各种情况进行实时动态调整,以保证整体云平台的可靠性、稳定性和可扩展性。

3 任务调度算法

3.1 主要任务调度算法

任务调度算法属于 NP-hard 问题^[11],并且由于在云环境中一些性能指标相互对立,无法实现对全部指标的一致优化,因此一般采用近似最优解的方式来取得一个较为平衡的调度结果。

通过不同方式可以对调度算法进行分类。根据任务类型的不同,可以将调度算法中的任务分为独立式和非独立式;根据任务运行环境的不同,可以将调度算法的运行环境分为同构、异构、静态、动态、集中和分布式等类型;根据任务调度算法的特点,可以将算法类型分为传统改进算法、启发式算法和

其他智能算法;根据任务调度的目标,可以将算法目标分为以性能为主、以服务质量为主、以利用率和能耗为主和以经济效益为主等几类。图 3 显示了任务调度算法中采用不同方式进行分类的结果。

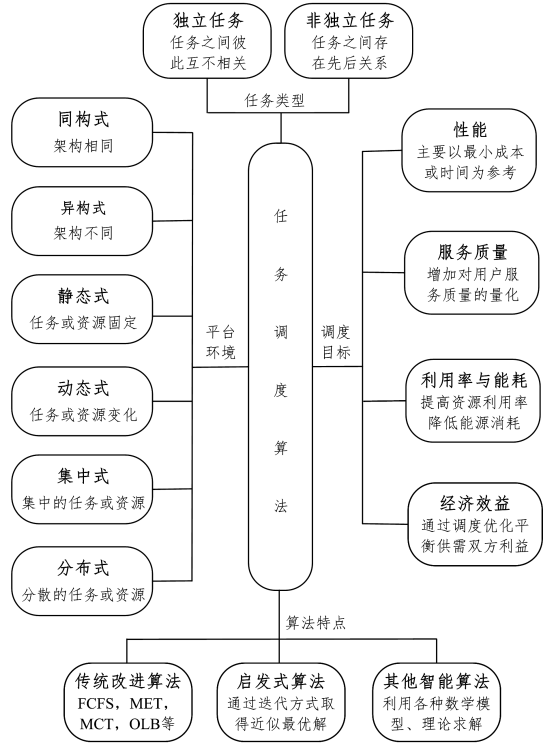


图 3 不同任务调度算法的分类

Fig. 3 Classification of different task scheduling algorithms

而具体来说,传统的任务调度算法有先来先服务(First Come First Served, FCFS)、最小执行时间(Minimum Execution Time, MET)、最小完成时间(Minimum Completion Time, MCT)、Max-Min 和 Min-Min^[12]等,但这些算法常常存在一定的局限性,不能直接作为云计算中的任务调度算法进行实际应用。结合云环境的特点,蚁群算法(Ant Colony Optimization, ACO)^[13]、遗传算法(Genetic Algorithm, GA)^[14]、粒子群算法(Particle Swarm Optimization, PSO)^[15]和模拟退火算法(Simulated Annealing, SA)^[16]等被先后应用到云计算中。这些算法一般是通过对解进行迭代优化后得出最终近似最优解,算法中参数阈值的设置以及自身的复杂度直接影响着算法的实际性能。表 1 中列出了一些常见的任务调度算法。

表 1 常见的任务调度算法

Table 1 Common task scheduling algorithms

算法名称	调度机制	优点	缺点
先来先服务	按照任务到达的先后顺序来调度资源	实现简单易行	性能有限
最小执行时间	按照任务的最短执行时间来调度资源	任务短、响应快	分配不均衡
最小完成时间	以任意顺序将任务分配到具有最早完成时间的资源上	任务完成时间短	整体运行时间长
Max-Min	将最大完成时间的任务分配到最早完成时间的资源上	实现简单,执行快速	缺乏可扩展性
Min-Min	将最小完成时间的任务分配到最早完成时间的资源上	实现简单,执行快速	负载均衡性能低
蚁群算法	模仿自然中的蚂蚁觅食方式,依据信息素获取最短路径	效率高,具有较强的鲁棒性	初期求解速度慢,搜索时间较长
遗传算法	模仿自然生物进化过程,以遗传、变异和交叉等方式不断迭代以获取更优解	易实现并行化,利于全局择优	效率低,稳定性差,容易过早收敛
粒子群算法	利用群体中个体对信息的共享,使得群体运动从无序到有序演化,并获得近似最优解	简单易实现,精度高,收敛快	容易早熟收敛,易陷入局部最优
模拟退火算法	利用固体退火原理模拟组合优化问题,从初始值开始迭代表减得到近似最优解	计算简单,鲁棒性强	收敛速度慢,执行时间长

3.2 任务调度性能评价参数

由图3可知,在云环境下,任务调度是一个需要综合考虑各种情况的复杂问题,只有根据实际需求选择恰当的调度算法才能够得到良好的效果,而实际需求又依赖于时间、成本、负载均衡、资源利用率、能效比等不同指标,这些也可视为云计算任务调度的性能评价参数。

(1) 执行时间

执行时间指任务从分配至资源到完成的运行时间。一个好的调度算法不仅要最小化每个任务的执行时间,更需要尽力缩短任务的总执行时间。

(2) 完成时间

完成时间可以描述为任务从提交开始到执行完成的整个调度时间,包括任务的总执行时间和延误时间,对用户来说就是从提交任务到获得结果的时长。

(3) 截止时间

截止时间是任务要求的最终完成时间。用户有特定需求时,可以在提交任务时设定对应的截止时间以保证任务按期完成。对于那些有很高时效性要求的任务来说,截止时间是调度算法中一个必不可少的关键指标。

(4) 成本

从用户的角度来说,成本是指用户为使用云平台而向云供应商所付出的费用。成本的高低通常与用户在云环境中使用的资源性能和运行时长有关。当采用成本限制时,调度策略将在用户的预算内实现任务完成时间的最小化^[17]。而对云供应商来说,成本是指在硬件设施、运行和维护等方面所投入的资金、人力等。

(5) 负载均衡

负载均衡指任务在不同虚拟机上、虚拟机在不同节点上的分配调度。在云环境中,虚拟机一般是运行任务的基本单元,在进行任务调度时,需要通过负载均衡来保证大多数时刻没有虚拟机或节点存在低载或过载的情况。负载均衡对改善调度算法的性能具有重要意义,不仅提高了资源的利用率,还能够通过低载资源的迁移和合并降低整体能耗^[18]。

(6) 资源利用率

资源利用率是指云环境中CPU、内存、存储、网络和应用等各类软硬件资源的使用效率。对云供应商来说,通过有限的设备尽可能地提高利用率能够为其带来更多的利益。

(7) 能效比

能效比反映了云环境中的能源利用情况,常与资源的计算性能和利用率有关。例如,当虚拟机或节点的CPU没有被充分利用时,能效比将由于计算性能的闲置而降低;而资源的超载运行同样会增大能耗,影响云计算的整体性能^[19]。

(8) 吞吐量

吞吐量是指在给定的一段时间内,云环境中不同虚拟机或物理机之间的数据传输量。调度算法需要考虑带宽的动态分配,以提高云计算的性能。

(9) 安全性

安全性是指云环境中各类数据的安全与隐私防护。由于

云计算通常是在一个分布式异构的环境下进行的,如何在数据传输与存储的过程中确保数据的一致性、可用性和隐私性,始终是一个复杂和关键的挑战。

(10) 可靠性

可靠性是指任务被成功运行并最终完成的概率。资源失效、硬件故障和网络堵塞都有可能造成任务失败,因此在任务调度时需要通过资源监控、任务重置等有效手段及时处理未能正常执行的任务或虚拟机,从而提高云计算平台的可用性和稳定性。

4 任务调度算法

云计算的概念被提出后,对任务调度算法在云环境中的研究和应用随着云计算技术的发展而不断推进。由于传统的任务调度算法不适用于云计算的特点,很多研究人员一方面对传统算法进行改进,另一方面又提出适合云环境的任务调度算法。云计算中的任务调度情况复杂多变,需要考虑的因素多种多样或是互为矛盾,因此云计算中的任务调度算法通常是针对某些性能指标进行提升和改善。下面将结合上述几种主要的评测指标对近年来国内外学者所提出的任务(资源)调度算法进行讨论。

4.1 不同的任务调度算法

4.1.1 面向时间的调度算法

文献[20]在混合云建设和管理项目中引进了基于二进制整数的算法来解决截止时间限制的任务调度优化问题。该算法在公有云下能够实现最小成本的调度,但在混合云环境中缺乏一定的鲁棒性,当任务复杂、完成时间较长时调度效率将急剧下降。

文献[21]在SaaS环境中采用一种局部关键路径算法(SC-PCP)来分配和调度大规模 workflow。该算法采用了递归和非最快速度等方式来降低成本,牺牲了一定的时间性能,同时不太适用于复杂的任务调度环境。

文献[22]基于混合云提出了一种预算限制包任务调度算法(BaTS),根据用户的成本要求在不同云资源中调度分配任务,实现成本和完成时间的最小化。该算法在任务调度过程中根据时间和成本动态调整集群机器,带来了额外的性能开销和网络传输;此外,该算法在最小化完成时间和提高灵活性等方面仍存在一定的改进空间。

4.1.2 面向成本的调度算法

文献[23]面向混合云提出了基于成本的任务调度技术来减少计算开销、数据传输和带宽限制,通过成本、截止时间和计算效率等方面来评价算法性能,并分析估值偏差对性能的影响。该算法采用最早截止时间优先方式实现成本的最小化,对于实时性要求较高的应用会产生多次调度和排序情况,造成性能的快速下降。

文献[24]构建了一个满足服务水平协议约束的用户需求模型,并基于基因算法提出了成本敏感的服务请求调度方法(CSRSGA)。该算法在SaaS应用中的SLA偏离、资源利用率和收益等方面的效果良好;但需要提前获知任务的执行时

间,不适用于实时调度环境。

文献[25]提出了一种解决云环境下大型程序运行需求调度的算法,该算法先使用帕累托优势分配任务到具有最大成本效益的虚拟机,再减少非关键任务的花费,从而实现整体成本和完成时间的最小化。该算法采用了成本效益因子来计算任务成本和完成时间,但缺乏在调度过程中对因子值的动态调整,算法的灵活性较低。

4.1.3 面向负载均衡的调度算法

文献[26]提出了一种自动代理负载均衡算法(A2LB)。通过负载计算、负载通道和负载迁移3个步骤,按照数据中心的负载情况进行虚拟机的动态调节,这种使用软件代理进行负载均衡的方式提高了资源利用率;但迁移过程中在可用虚拟机的搜索上耗时过多。

文献[27]基于模拟退火算法提出了一种云环境下的负载均衡调度方法(SALB)。该算法根据物理设备的CPU利用率进行虚拟机的动态迁移。该算法具有比传统模拟退火算法和轮转算法更好的负载调度性能;但容易造成较高的迁移成本,同时也未考虑内存、带宽等其他负载指标。

文献[28]提出了一种结合模糊理论和萤火虫群优化的负载均衡模型(FUZZY AND GSO),该模型根据模糊理论,按照负载和计算能力选择虚拟资源,并根据萤火虫群优化方法迁移高负载主机上的虚拟机到其相邻主机。该模型具有较高的资源利用率和适应能力;但在进行虚拟机迁移时耗时较多,缺乏对任务成本和时间等指标的考虑。

4.1.4 面向资源利用率的调度算法

文献[29]基于超启发式算法,依据细菌觅食优化技术提出了一种资源调度算法(BFOHH),该算法通过卢布林模型形成 workflow,并按照预期时间计算模型来比较执行时间和成本的变化。该算法弥补了有关算法在异构环境中面对任务时性能不足及适应性差等缺点;但在进行资源调度时缺乏对可用节点的优化管理,资源的稳定性和可靠性仍需进一步提高。

文献[30]将分组和动态优化技术结合起来进行云环境中的资源调度,通过优化后的任务分组调度算法减少处理时间和成本,再根据资源的处理性能和开销动态分配虚拟机。该调度方法采用贪婪方式,在满足任务成本的基础上动态优化资源分配,兼顾了用户和云供应商双方的利益;但未充分考虑任务和资源的整体优化,调度性能有限。

文献[31]提出了一种自适应混合启发式调度算法,即在用户成本和截止时间的约束下使用遗传算法初次匹配任务到资源,再通过动态关键路径调度算法(DCP-G)实时调整任务级别。该算法通过 workflow 图中的关键路径,在任务和资源之间进行有效匹配,对不同任务和资源类型都能获得较好的性能;但该算法对关键路径和执行时间等参数的实时更新增大了调度开销,同时也缺乏对资源负载和能耗等方面的考虑。

4.1.5 面向能耗的调度算法

文献[32]提出了一种基于服务水平协议的能源重配软件系统(EASY)来减少CPU的使用率。该算法通过网络性能队列模型和在线算法动态调整进程,从而达到能源消耗最小

的目的。该软件通过启发式技术解决节能所对应的混合整数规划问题,加快了问题求解的响应速度;但该方法无法影响处于非可操作状态的云节点,且容易造成性能瓶颈和单点失效。

文献[33]描述了一种基于控制依赖图的能耗任务调度算法(EES),在分析和计算最差性能后通过扩展传统多处理器调度算法来实现分布式环境中的最小能源消耗。该算法通过单位成本和定价策略将任务调度到合适的资源节点,执行完任务后休眠或关闭节点。这种定价调度方式使得用户和云供应商都能获得最大利益;但缺乏对整体调度的有效规划和实时调整,同时未能充分提高云环境中的资源利用率。

文献[34]提出了基于粒子流算法的动态电压和频率调节调度算法(DVFS-MODPSO),该算法在平衡调度性能和能耗的基础上通过调整时钟频率来减少不同负载电压上的电力消耗。其为用户提供了多种可行方案以满足成本、时间等不同需求;但仍缺乏对可靠性和安全性等方面的考虑,且在负载调整方面仍有较大的改进空间来进一步降低能耗。

文献[35]面向整体负载提出了基于地理分布的数据中心调度平衡策略,通过结合绿色能源和冷电力设计出 MinBrown 调度技术来提高能源利用率。该调度方案通过计算任务在不同数据中心上所消耗的能源情况进行任务调度。与其他调度方式相比,这种方案较大地减少了能源消耗;但对数据中心的带宽、环境和能源供应提出了一定的要求。

4.1.6 面向稳定性的调度算法

文献[36]提出了一种预见遗传算法(LAGA)来优化分布式系统中任务流的完成时间和稳定性,所建立的指数失败模型通过计算资源的失败概率来最大化系统的调度可靠性。该算法根据反映任务资源匹配优先策略的遗传因子进行性能的评价;但其整体实现过程略显复杂,各种参数选择不当时容易影响调度的性能。

文献[37]首先提出了具有不同服务质量要求的模型,然后采用蚁群算法来优化服务流程调度。该算法采用了违约率来描述云供应商对服务水平协议的违背,并通过相应模块监控云服务的运行状态。该算法综合考虑了稳定性、响应时间、成本和安全性等指标;但其对异构环境的适应能力较弱,当任务类型和云服务数量发生较大变化时,其调度性能下降明显。

文献[38]对云环境下用户和虚拟机的安全性进行了量化,通过定量转换模型对安全概念进行了数值化表述,并基于离散粒子流算法构建了满足多维度服务质量要求的调度算法(CWDPSO)。该算法对安全性、完成时间和成本3个指标进行了调度优化,增强了云环境的稳定性;但其在迭代过程中对虚拟机和任务安全优先级的判定会对调度性能造成一定影响。

4.2 任务调度算法特点的对比

由于不同调度算法的侧重点不同,因此很难用同一种标准衡量出各种算法的性能优劣。为综合分析和相互比较,下面从应用策略、算法特点和侧重点等方面对近年来的任务(资源)调度算法进行了简要概括和分类对比。表2从算法类型、基本流程、参数和特点等方面进行了文献分析;图4则按照优化目标对文献[20-48]进行了归纳。

表2 一些云计算任务调度算法的概要分析

Table 2 Summary analysis of some task scheduling algorithms in cloud computing

策略	算法名	算法描述	参数	工具	特点
启发式	基于动态电压和频率调节的节能调度算法 ^[39]	该算法通过能耗比(ECR)来评价虚拟机在不同频率下在CPU中执行任务的效率,尽可能将任务分配给活动中的虚拟机,同时调整对应CPU频率以节约能源,并在任务结束后通过动态迁移以实现负载均衡。	任务执行时间、任务需求频率、能源消耗、任务权重	Dell R720 servers NAS Parallel Benchmarks 3.3 (NPB)	优点:实时动态调节 缺点:计算任务的能耗比在一定程度上影响了性能
智能算法	基于动态负载调节的自适应任务调度算法 ^[40]	该算法针对任务调度在Hadoop集群中不能自动调节和实时负载等问题,提出了ATSDWA调度算法。根据集群间心跳间隔判断CPU、内存和负载的使用率,并与设定的任务上下限值相比较后进行调度分配。	集群节点的CPU最大和最小核数、最大任务初始量、心跳时间	VMware virtual machine(VM) platform, Hadoop	优点:效率和可用性较高,吞吐率提高 缺点:适用性较差,集群规模有限
Min-Min	基于能耗优化的任务调度算法 ^[41]	该算法根据服务器能源模型和服务器性能模型较精确地对服务器在不同负载下的能耗效率进行评估,并经过基于能耗的任务调度算法(ECOTS)得出能效矩阵,采用能效最大化方式将任务分配到对应的服务器上。	服务器和任务信息、任务能效值	A simulation framework	优点:算法时间和空间复杂度低,能效比高 缺点:负载均衡效果差,应用范围有限
遗传算法	一种改进的遗传调度算法 ^[42]	该算法(N-GA)将基因(调度任务)按优先权降序排列作为染色体,通过3个启发式等级策略初始化种群,再按照80%的比例进行交叉后单点变异。该算法分别通过行为模型和线性时态逻辑范式进行了正确性和期望值演算,并结合NuSMV和PAT模型做了进一步验证。	任务平均计算成本、任务互通权值、处理器执行和完成时间	Azure cloud	优点:完成时间少,稳定性高 缺点:限于静态任务,计算较为复杂
智能算法	一种面向预算限制的任务调度长度最小化算法 ^[43]	该算法(MSLBL)将一组任务的整体预算限制分解到每个任务上,分别计算出各任务的预计、实际以及预算标准,并在执行费用不超出预算的前提下,按照任务优先级,以最小完成时间为原则,将任务调度到对应的处理器。	任务执行费用、处理器单位价格、预算约束	A simulator in Java language	优点:成本可控,完成时间较短 缺点:缺乏动态调节和负载均衡等约束
群体智能算法	基于猫群优化和正交田口方法的任务调度算法 ^[44]	该算法(OTB-CSO)采用正交田口方法对猫群算法中的跟踪模式进行优化处理,通过随机扰动产生两个速度集,并依照田口正交阵列矩阵来更新跟踪模式下猫的位置。	结合率、记忆池、变化域、变化数	CloudSim	优点:最小化完成时间,负载均衡性好 缺点:缺乏对能效和成本等多目标的优化
机器学习	基于强化学习的任务调度算法 ^[45]	该算法由任务调度子模型(TSSM)、任务执行子模型(TESM)和任务传递子模型(TTSM)组成一个排队模型,在分析每个子模块的响应时间后采用强化学习方法(RL)进行任务调度并最小化模型响应时间。	当前状态、操作集、折扣因子、状态转移函数、强化函数	MATLAB CloudSim	优点:响应速度快,资源利用率高 缺点:缺乏动态调节,调度稳定性一般
贪婪算法	一种支持软实时应用的调度算法 ^[46]	该算法提出了一个符合QoS需求的模糊资源分配框架来满足用户的软实时调度,以每一阶段的首次最大赤字(LDF)依次调度用户任务,其中用户的优先级根据历史事件动态调整。	赤字向量、优先级、权重值、期望完成向量	A Simulation Platform	优点:资源利用率高,满足异构环境 缺点:实时约束不足,适用性较弱
启发式	一种限制服务延迟的利润最大化任务调度算法 ^[47]	该算法(PMA)根据公有云的虚拟机执行价格动态调整私有云的单元任务价格,以便最大化利润,并使用模拟退火粒子群优化算法(SAPSO)获得无约束条件下的单位时间利润。	服务延迟和任务平均执行时间、任务执行价格、云处理器执行价格	Google compute systems	优点:利润最大化,高吞吐量,动态价格 缺点:未考虑执行时间和负载均衡等约束
智能算法	一种弹性监控和增强可信的任务调度算法 ^[48]	该算法(GAMESH)将大规模的完全分布式系统按监控层和控制层、内部层和外部层进行划分,通过数据弹性分布监控(DCM)和任务执行失败排错(DCJT)实现高效的任务调度管理。	计算能力、负载域、校正因子、影响节点数	A real geographically distributed data centers across Europe	优点:高吞吐量,强扩展性 缺点:调度效率一般,任务成本较高

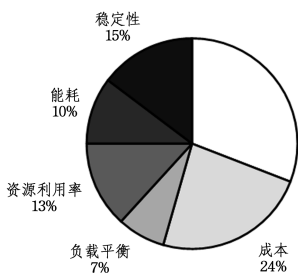


图4 相关文献中的侧重点分布

Fig. 4 Focus of task scheduling in related literatures

4.3 任务调度算法的分析

前文对近年来的一些云计算任务调度文献进行了调研汇总,虽然所调研文献数量有限,存在一定的局限性,但不难发现,目前在该领域的研究仍处于初始阶段。从算法策略和评价指标等方面的对比中可以看出,云计算中的任务调度还存在不少困难有待解决。

(1) 调度任务和平台环境

不同于传统系统的稳定任务和静态环境,在云计算中任务调度所面临的业务流和可用资源是实时变化的,初始时调度程序通常无法获取全部任务和所有资源信息,需要将任务根据时间、成本等因素实时分配到异构资源上,同时还应能够根据云环境的整体动态变化合理调度资源,在确保用户数据安全和系统稳定的前提下尽可能达到各性能指标间的最佳平衡。目前,多数文献中所提出的调度算法仍是基于一个相对稳定的云环境,较少综合考虑各种任务类型以及调度的分布式、动态式和异构式环境。

(2) 服务质量和水平协议

大多数的任务调度算法都会从一种或多种性能评测指标出发,通过评测服务水平协议来确保满足一定的用户服务质量;但在进行服务水平协议偏离监控时,难免造成调度程序的额外开销,从而影响到整体算法和云平台的性能。因此,调度算法在保证用户服务质量的同时,还需要

减少或避免额外的性能开销。

(3)资源利用和能源消耗

考虑到云计算所涉及的硬件规模动辄以万记,针对每个硬件的微弱性能调整都会给整体平台带来巨大影响,因此合理分配各类资源,提高利用率并降低能耗始终是云计算中的一个关键环节,这也是调度算法的重要研究方向之一。其意义不仅在于降低云平台的运行成本,更是当前解决环境污染和节能减排的一种有效手段。然而,云计算性能的提升又常常会导致能源开销的增加,因此调度算法需要考虑如何在提升性能和降低能耗之间取得平衡。基于资源利用率和能耗的研究将持续成为业界关注的重点。

(4)单一算法和复合算法

云计算中任务调度的特点和各类指标的性质,使得难以有某种算法能够同时完全满足各种需求。目前存在的调度算法往往只能针对调度中的某几项指标进行优化,因此在越来越多的研究中采取了复合算法,即将两种或多种算法结合在一起,分别提供不同侧重点的优化。然而,多种算法的结合将直接增加调度程序的复杂程度,进而影响到整体调度性能。如何在保证性能的前提下尽可能地降低算法复杂度,亦是云计算任务调度中需要不断研究的方向之一。

结束语 云环境中,在大规模的不同资源上进行实时任务调度是一项巨大的挑战,目前尚未有一种完全有效的技术方案来解决这一难题。基于此,本文根据云计算中任务调度的特点,结合近年来国内外的相关研究,对目前不同的任务调度技术进行了分类归纳;结合主要评价参数综合对比了有关算法的特点,并对相关文献中所提到的算法进行了概要汇总。从中可以看出,目前各种云计算任务调度算法所采用的技术方案和性能指标各有不同,尚没有一种完善的解决途径。这一方面是由于云计算中的任务调度与传统方式大不相同,其复杂度和困难性远远超过了单一模式;另一方面是因为调度算法在云环境中所要考虑的性能指标很多,难以同时实现,甚至有些指标间存在相互冲突的情况。因此,一种好的调度算法需要根据任务和资源的不确定性和动态变化性灵活采取调度策略,并尽可能多地考虑性能指标,在一定范围内进行平衡,综合提高云计算的运行效率和服务质量。

参 考 文 献

- [1] KEAHEY K, FIGUEIREDO R, FORTES J, et al. Science clouds; Early experiences in cloud computing for scientific applications[C]// Proceedings of Cloud Computing and Its Applications. 2008; 825-830.
- [2] TSAI J T, FANG J C, CHOU J H. Optimized task scheduling and resource allocation on cloud computing environment using improved differential evolution algorithm[J]. Computers & Operations Research, 2013, 40(12): 3045-3055.
- [3] LIN W W, QI D Y. Survey of Resource Scheduling in Cloud Computing[J]. Computer Science, 2012, 39(10): 1-6. (in Chinese)
林伟伟, 齐德昱. 云计算资源调度研究综述[J]. 计算机科学, 2012, 39(10): 1-6.
- [4] VIJINDRA, SHENAI S. Survey on Scheduling Issues in Cloud Computing[J]. Procedia Engineering, 2012(38): 2881-2888.
- [5] ZHAN Z H, LIU X F, GONG Y J, et al. Cloud Computing Resource Scheduling and a Survey of Its Evolutionary Approaches[J]. Acm Computing Surveys, 2015, 47(4): 1-33.
- [6] PANWAR N, RAUTHAN M S. Analysis of various task scheduling algorithms in cloud environment: Review[C]// International Conference on Cloud Computing, Data Science & Engineering-Confluence. IEEE, 2017: 255-261.
- [7] ISRAR H. Architecture level mapping of Cloud Computing with Grid Computing[J]. International Journal of Engineering Sciences & Emerging Technologies, 2013, 5(1): 7-11.
- [8] FOSTER I, ZHAO Y, RAICU I, et al. Cloud Computing and Grid Computing 360-Degree Compared[C]// Grid Computing Environments Workshop. IEEE, 2009: 1-10.
- [9] KOMARASAMY D, MUTHUSWAMY V. Adaptive Deadline Based Dependent Job Scheduling algorithm in cloud computing[C]// Seventh International Conference on Advanced Computing. IEEE, 2016: 1-5.
- [10] BUYYA R, GARG S K, CALHEIROS R N. SLA-oriented resource provisioning for cloud computing: challenges, architecture, and solutions[C]// International Conference on Cloud and Service Computing. IEEE Computer Society, 2011: 1-10.
- [11] WU Q W, ISHIKAWA F, ZHU Q S, et al. Deadline-Constrained Cost Optimization Approaches for Workflow Scheduling in Clouds[J]. IEEE Transactions on Parallel & Distributed Systems, 2017, 28(12): 3401-3412.
- [12] BIN X L. Research on The Scheduling Technology of Tasks in Real-Time Systems[D]. Changsha: National University of Defense Technology, 2004. (in Chinese)
宾雪莲. 实时系统中的任务调度技术研究[D]. 长沙: 国防科学技术大学, 2004.
- [13] DORIGO M, CARO G D. Ant colony optimization: a new metaheuristic[C]// Proceedings of the 1999 Congress on Evolutionary Computation, 1999(CEC 99). IEEE, 2002: 1470-1477.
- [14] 王小平, 曹立明. 遗传算法: 理论、应用与软件实现[M]. 西安: 西安交通大学出版社, 2002: 10-50.
- [15] BRATTON D, KENNDY J. Defining a standard for particle swarm optimization[C]// IEEE Swarm Intelligence Symposium. Honolulu, 2007: 35-46.
- [16] RODRIGUEZ F J, GARCIA-MARTINEZ C, LOZANO M. Hybrid Metaheuristics Based on Evolutionary Algorithms and Simulated Annealing: Taxonomy, Comparison, and Synergy Test[J]. IEEE Transactions on Evolutionary Computation, 2012, 16(6): 787-800.
- [17] JIA Y, BUYYA R. Scheduling scientific workflow applications with deadline and budget constraints using genetic algorithms[J]. Scientific Programming, 2006, 14(3-4): 217-230.
- [18] ORGERIE A, LEFÈVRE L, GELAS J P. Save Watts in Your Grid; Green Strategies for Energy-Aware Framework in Large Scale Distributed Systems[C]// IEEE International Conference on Parallel and Distributed Systems. IEEE Computer Society, 2008: 171-178.
- [19] WANG X L, WANG Y P, ZHU H, et al. Energy-Efficient Multi-Job Scheduling Model for Cloud Computing and Its Genetic Algorithm[J]. Mathematical Problems in Engineering, 2014, 2012(6): 152-170.

- [20] BOSSCHE R V D, VANMECHELEN K, BROECKHOVE J. Cost-Optimal Scheduling in Hybrid IaaS Clouds for Deadline Constrained Workloads[C]// IEEE International Conference on Cloud Computing. IEEE Computer Society, 2010; 228-235.
- [21] ABRISHAMI S, NAGHIBZADEH M. Deadline-constrained workflow scheduling in software as a service Cloud[J]. *Scientia Iranica*, 2012, 19(3): 680-689.
- [22] OPRESCU A M, KIELMANN T. Bag-of-Tasks Scheduling under Budget Constraints[C]// IEEE Second International Conference on Cloud Computing Technology and Science. IEEE Computer Society, 2010; 351-359.
- [23] BOSSCHE R V D, VANMECHELEN K, BROECKHOVE J. Online cost-efficient scheduling of deadline-constrained workloads on hybrid clouds[J]. *Future Generation Computer Systems*, 2013, 29(4): 973-985.
- [24] LIU Z P, WANG S G, SUN Q B, et al. Cost-Aware Cloud Service Request Scheduling for SaaS Providers[J]. *Journal of Beijing University of Posts & Telecommunications*, 2013, 57(2): 291-301.
- [25] SU S, LI J, HUANG Q J, et al. Cost-efficient task scheduling for executing large programs in the cloud[J]. *Parallel Computing*, 2013, 39(4-5): 177-188.
- [26] SINGH A, JUNEJA D, MALHOTRA M. Autonomous Agent Based Load Balancing Algorithm in Cloud Computing[J]. *Procedia Computer Science*, 2015, 45: 832-841.
- [27] FAN Z Q, SHEN H, WU Y B, et al. Simulated-Annealing Load Balancing for Resource Allocation in Cloud Environments[C]// International Conference on Parallel and Distributed Computing, Applications and Technologies. IEEE, 2014; 1-6.
- [28] SINGHAL U, JAIN S. A New Fuzzy Logic and GSO based Load balancing Mechanism for Public Cloud[J]. *International Journal of Grid & Distributed Computing*, 2014, 7(5): 97-110.
- [29] ARON R, CHANA I. Bacterial foraging based hyper-heuristic for resource scheduling in grid computing[J]. *Future Generation Computer Systems*, 2013, 29(3): 751-762.
- [30] CHAWLA Y, BHONSLE M. Dynamically Optimized Cost Based Task Scheduling in Cloud Computing[J]. *International Journal of Emerging Trends & Technology in Computer Science*, 2013, 2(3): 38-42.
- [31] RAHMAN M, HASSAN R, RANJAN R, et al. Adaptive workflow scheduling for dynamic grid and cloud computing environment[J]. *Concurrency & Computation Practice & Experience*, 2013, 25(13): 1816-1842.
- [32] MARZOLLA M, MIRANDOLA R. Dynamic power management for QoS-aware applications[J]. *Sustainable Computing Informatics & Systems*, 2013, 3(4): 231-248.
- [33] MA Y, GONG B, SUGIHARA R, et al. Energy-efficient deadline scheduling for heterogeneous systems[J]. *Journal of Parallel & Distributed Computing*, 2012, 72(12): 1725-1740.
- [34] YASSA S, CHELOUAH R, KADIMA H, et al. Multi-objective approach for energy-aware workflow scheduling in cloud computing environments[J]. *The Scientific World Journal*, 2013, 2013(3-4): 1-13.
- [35] CHEN C B, HE B S, TANG X Y. Green-aware workload scheduling in geographically distributed data centers[C]// IEEE International Conference on Cloud Computing Technology and Science. IEEE, 2013; 82-89.
- [36] WANG X, YEO C S, BUYYA R, et al. Optimizing Makespan and Reliability for Workflow Applications with Reputation and Look-ahead Genetic Algorithm[J]. *Future Generation Computer Systems*, 2012, 27(8): 1124-1134.
- [37] LIU H, XU D, MIAO H K. Ant Colony Optimization Based Service Flow Scheduling with Various QoS Requirements in Cloud Computing[C]// First ACIS International Symposium on Software and Network Engineering. IEEE Computer Society, 2011; 53-58.
- [38] CAO J F, CHEN J J, ZHAO Q S. An optimized scheduling algorithm on a cloud workflow using a discrete particle swarm[J]. *Cybernetics & Information Technologies*, 2014, 14(1): 25-39.
- [39] WANG S Y, QIAN Z Z, YUAN J B, et al. A DVFS Based Energy-Efficient Tasks Scheduling in a Data Center[J]. *IEEE Access*, 2017, 5(99): 13090-13102.
- [40] XU X L, CAO L L, WANG X H. Adaptive Task Scheduling Strategy Based on Dynamic Workload Adjustment for Heterogeneous Hadoop Clusters [J]. *IEEE Systems Journal*, 2017, 10(2): 471-482.
- [41] LIN W W, WANG W Q, WU W T, et al. A Heuristic Task Scheduling Algorithm Based on Server Power Efficiency Model in Cloud Environments [J/OL]. <https://doi.org/10.1016/j.suscom.2017.10.007>.
- [42] KESHANCHI B, SOURI A, NAVIMPOUR N J. An improved genetic algorithm for task scheduling in the cloud environments using the priority queues; Formal verification, simulation, and statistical testing[J]. *Journal of Systems & Software*, 2017, 124(2017): 1-21.
- [43] CHEN W H, XIE G Q, LI R F, et al. Efficient task scheduling for budget constrained parallel applications on heterogeneous cloud computing systems[J]. *Future Generation Computer Systems*, 2017, 74(C): 1-11.
- [44] GABI D, ISMAIL A S, ZAINAL A, et al. Orthogonal Taguchi-based cat algorithm for solving task scheduling problem in cloud computing [J]. *Neural Computing & Applications*, 2016(1): 1-19.
- [45] PENG Z P, CUI D L, ZUO J L, et al. Random task scheduling scheme based on reinforcement learning in cloud computing[J]. *Cluster Computing*, 2015, 18(4): 1595-1607.
- [46] DU Y H, VECIANA G D. Scheduling for Cloud-Based Computing Systems to Support Soft Real-Time Applications [C] // IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications. IEEE, 2016; 1-9.
- [47] YUAN H T, BI J, TAN W, et al. Temporal Task Scheduling With Constrained Service Delay for Profit Maximization in Hybrid Clouds[J]. *IEEE Transactions on Automation Science & Engineering*, 2017, 14(1): 337-348.
- [48] BELLAVISTA P, CINQUE M, CORRADI A, et al. GAMESH: A grid architecture for scalable monitoring and enhanced dependable job scheduling[J]. *Future Generation Computer Systems*, 2016(71): 192-201.