CN 43-1258/TP
ISSN 1007-130X

计算机工程与科学
Computer Engineering & Science

第 41 卷第 9 期 2019 年 9 月
Vol. 41, No. 9, Sep. 2019

文章编号:1007-130X(2019)09-1525-09

# 用于高性能计算的作业调度能效性研究综述*

郑文旭,潘晓东,马　迪,汪　浩

(国防科技大学计算机学院,湖南 长沙 410073)

**摘　要**:由于科学研究与商业应用等对高性能计算的需求与日俱增,高性能计算的性能和系统规模得到迅速发展。但是,急剧增长的功耗严重限制了高性能计算系统的设计和使用,使得低功耗技术成为高性能计算领域的关键技术。作为整个系统的核心组件,作业调度系统立足有限的系统资源,对用户提交的应用进行作业-资源分配,其能效性对于整个高性能计算系统的能耗控制与调节起到至关重要的作用。首先介绍主要的能量效率技术和常用的作业调度策略,然后对当前高性能计算作业调度能效性进行分析,并讨论了其面临的挑战及未来发展方向。

**关键词**:高性能计算;作业调度;能量有效性

**中图分类号**:TP301　　**文献标志码**:A

**doi**:10.3969/j.issn.1007-130X.2019.09.001

# Overview on the energy efficiency of job scheduling for high performance computing

ZHENG Wen-xu, PAN Xiao-dong, MA Di, WANG Hao

(School of Computer, National University of Defense Technology, Changsha 410073, China)

**Abstract**: Due to the increasing demand for high performance computing in scientific research and commercial applications, the performance and system scale of high performance computing are developing rapidly. However, the rapid increase in power consumption severely limits the design and use of high-performance computing (HPC) systems, which makes low-power technologies become the key technology in the HPC field. As the core component of the whole system and based on limited system resources, the job scheduling system distributes job resources to the applications submitted by the users. Its energy efficiency plays an important role in the control and regulation of the energy consumption of the whole HPC system. We first introduce the main energy efficiency techniques and common job scheduling strategies, then analyze the energy efficiency of current HPC job scheduling, and discuss the challenges it faces and future development directions.

**Key words**: high performance computing (HPC); job scheduling; energy efficiency

## 1　引言

过去几十年中,高性能计算 HPC(High Performance Computing)在能源、生物、气象、科研、地质勘探等计算密集型应用中得到长足发展。随着应用程序趋向于大型化、并行化,需要更高性能的分布式计算系统,而系统性能的快速提升又带来组件密度的增加,能耗问题随之引起人们普遍关注。IBM 发布报告[1]指出实现百亿亿次计算系统的 5 大瓶颈,其中能耗瓶颈排在首位。2014 年美国能源局更是将能量和功耗问题作为未来 E 级系统面临的首要挑战[2]。根据“功耗摩尔定律”,每隔 18 个月计算机结点功耗就会翻倍[3]。未来更大规模

* 收稿日期:2018-10-29;修回日期:2019-03-18
基金项目:国家数值风洞项目(NNW2018-ZT6B13)
通信地址:410073 湖南省长沙市国防科技大学计算机学院
Address: School of Computer, National University of Defense Technology, Changsha 410073, Hunan, P. R. China

的高性能计算系统功耗更将不断攀升,其发展将面临严重的功耗危机,将会在一定程度上阻碍其快速发展。

Ge等人[4]研究了5台超级计算机,发现这些高性能计算系统的性能仅为峰值性能的54%~71%,在实际应用中,其性能仅为峰值的10%。如此低的性能比重主要是缘自于在庞大的集群中,各种计算、通信和I/O操作在结点之间的分布不均衡,导致部分结点运行速度相差太大,大量的松弛时间(Slack Time)造成了计算资源和能量的浪费。为了控制负载分配,作业调度策略在处理器数量有限的高性能计算系统中对用户提交的应用程序进行作业-资源分配[5],这对于合理利用计算结点并确保并行作业高效运行起到重要作用。

本文主要对高性能计算系统中作业调度能量有效性进行综述,能量有效性(Energy-Efficiency of Job Scheduling System,下文简称"能效性")为单位时间内结点负载与能量消耗之间的比值,用于衡量系统在作业调度过程中的能量效率。对于不断发展的高性能计算系统,能提供给用户作业负载运行的计算资源已经不再是设计并行作业调度策略的唯一考虑因素,由于庞大的系统引发的能耗问题日益突出,因此如何最大化作业调度能量有效性成为关键因素。

本文第2节介绍作业调度能效的研究背景,主要是被广泛运用于高性能计算系统中的动态电压频率调节DVFS(Dynamic Voltage and Frequency Scaling)和动态功耗管理DPM(Dynamic Power Management)两个传统的能量效率技术。

第3节对作业调度系统、算法进行了回顾,这些单纯的策略主要以提升资源利用率,减少资源碎片,提升作业吞吐率,减少饥饿作业为目的,并没有充分考虑作业在不同结点上运行时功耗的变化,也没有考虑作业的能效需求差异。由于没有足够的作业负载,以及作业调度和资源分配策略无法充分利用系统资源,常常会有空闲资源浪费系统能耗的情况。

作业调度策略研究领域的相关技术发展,融合了应用和系统层面,追求既能满足用户实时需求,又能最大化利用系统资源,从而在一定程度上提高作业调度能效性。第4节将介绍调度系统、调度算法能效性优化技术和自适应功耗管理策略等。这些方法在一定程度上提高了资源利用率,缩短了作业等待时间,提高了高性能计算系统总体性能,同时也兼顾了作业调度的能效性,取得了一定的积极效果。

当前,作业调度能效性的研究处于起步阶段,仍然有许多地方亟待研究人员深入研究。在第5节中,我们讨论了高性能计算作业调度能效性的主要挑战,以及未来应该关注的方向,包括:计算认知能力、设备异构性、调度公平性等方面。

## 2 背景

高性能计算主要在分布式集群或超级计算机上进行,对于传统的高性能计算应用,研究人员主要关注的是系统的性能、可靠性和安全问题,因此在高性能计算中节约能耗的问题并没有得到足够的关注[6-9]。近年来,随着现代数据中心能源需求的稳步增长,人们开始意识到能源消耗问题也是至关重要的。

起初,研究人员主要研究计算机系统内部对功率感知的处理器和内存设计技术,提出了静态与动态的功率管理方法[10],分别从热感知的硬件设计和功率感知的软件设计方面入手[11],以降低处理器和内存资源的能耗。文献[12-15]将能效性策略主要归结为:(1)动态电压频率调节DVFS技术;(2)动态功耗管理DPM技术;(3)提高服务器、存储器和冷却效率的技术;(4)多核处理器设计;(5)虚拟化技术。

由于在整个高性能计算系统中,处理器的功耗占总功耗的很大一部分(作业负载时,处理器能耗占整个结点能耗的将近50%[16];在超级计算机中,CPU支配着整个系统功耗的35%[17]),负载功率越高,能耗越大。控制CPU功耗的DVFS技术[18]能够通过降低电源电压或时钟频率来降低功耗,这已经成为一个热门的研究课题[19-22],被广泛应用于Intel Xeon、AMD Atholon和ATI co-processors等处理器中。越少的工作负载导致越少的能耗,这是能量比例技术的概念,DVFS就是这个技术的一个例子[23]。DVFS可以被分为2个层次[20]:行为层和系统层。在行为层中,一旦一个功能单元在设计期间确定了功率电压,它在运行过程中将保持不变;在系统层中,处理器的供电电压可以在运行过程中发生变化,这就为降低系统能耗提供了更大的灵活性和潜力。因此,利用DVFS进行功耗调整,主要是在系统层面上。Etinski等人[5]基于功率配置的整型线性方程提出了利用DVFS调整结点功耗的并行作业调度策略,从而实现对整个系统功耗的控制。虽然通过在较低频率/

电压下运行处理器能有效地减少总功耗,但也可能会以增加作业执行时间为代价[5,24,25]。为了缩短作业执行时间,研究者提出了在给定功耗阈值的情况下提高功率的方法(将在第4节中介绍),但也无法显著地节省功耗,更有可能给硬件带来一定的损伤。

随着系统结点的增多,结点间通信成为重要的能耗,尤其在大规模分布式计算系统中,通信引起的能耗是巨大的,有必要对应用与系统资源进行综合协调,以发挥最大效益。而在分布式的集群系统中,每个组件要么处于活动运行状态,要么处于不同的休眠或者断电状态。例如,在ACPI(Advanced Configuration and Power Interface)标准中[26],处理器的活动状态为$C0$,而休眠状态可以分为$C1$,$C2$,…,$Cn$,不仅仅是处理器,内存、芯片、磁盘、I/O总线和其它设备都有活动、就绪与休眠等多个不同状态[26-32]。Chen等人[33]研究表明,空闲状态下的元件消耗了整个系统中峰值负载电量的大约2/3,所有的功能元件在低负载或零负载时都会有大量的能耗。而事实上,一个数据中心资源的平均负载仅占总资源的30%[34],这就为调整大约70%资源的状态模式,从而减少空闲功耗提供了可能[35]。为了提高系统能效性,系统选择性地暂停设备或者调整空闲资源状态,当需要时才启动已停止工作的设备,以优化活动资源的数量,使其较准确地满足应用程序需求[16,24,27,36]。

DPM就是运用了这一项技术,动态地配置系统资源,为所运行的作业提供最小数量的活动组件或最小负载,以达到节能的效果[35,37]。DPM技术可以运用到CPU[38]、存储盘、内存[39]、服务器和网络设备[23,35,40]等不同的耗电元件,Benini等人[37]利用DPM方法实现了电子设备动态重新配置,为用户的服务请求提供了最小数量的活动部件,从而最小化系统能耗。

由于这种方法高度依赖于系统工作负载,因此该方法的关键挑战是确定何时关闭哪个结点或其组件[41]。而且不同设备存在不同的电气性能,关闭后启动设备或者转换状态时带来的额外功耗、过渡成本难以估计,可能会超过用户设定的功耗阈值,抵销了一个或多个低功耗操作状态带来的节能效果。例如,处理器休眠程度越深,能耗越少,但唤醒处理器或转变状态都需要更多的能量[28,42]。而且在作业调度策略中,关闭空闲结点或者状态转换并不是实时的,时效性较差[43,44],有悖于高性能计算对于运行速度的要求。

DVFS和DPM是目前研究比较广泛的两种系统层级的能效性策略。DVFS相对更细粒化,更加关注处理器和内存的协调[45],但也有系统级作业分配能效性的相关研究[46]。而DPM可以被认为是一种作业-资源调度策略,其未来发展更有可能是融合不同层次的,综合使用DVFS技术在通信计算阶段节省处理器功耗,同时在运行时将负载平衡技术用于结点内存的资源调度[47-51],进一步节省能耗。经过几十年的发展,研究人员已经提出了许多作业调度算法和系统[52-54],其关注点也逐渐从执行速度转移到能量效率上。

## 3 传统作业调度策略

高性能计算系统的能效性可以在3个层次上改善:(1)能效性的应用程序;(2)功率感知的资源管理系统;(3)高效率的硬件基础[46]。而作业调度就是在第2个层次上进行的。设计一个高效的作业调度系统并不是容易的事,因为不同的应用程序、工作负载、集群系统和调度策略的节能差异很大[16]。作业调度最主要的功能就是根据作业调度策略从作业队列中选择合适的作业,并通知资源管理器为其分配资源[17](如图1所示),主要关注运行作业所需要的资源和时间[39]。计算资源利用率和用户所提交的作业等待时间决定了调度策略的好坏,影响作业调度算法的因素包括了用户提交作业的序列分布、系统的利用率、单位时间内平均完成的作业数、作业的平均周转时间等。以往的高性能计算系统在设计调度策略时很少考虑能效性[37],但能效性优化往往正是从以上几个方面入手的,如图1虚线框内所示。本节主要从常用的先来先服务FCFS和回填BackFill策略入手,介绍传统的作业调度策略。

### 3.1 先来先服务策略

先来先服务FCFS(First Come First Serve)策略是一种简单的静态作业调度策略[55],按到达顺序运行作业,不允许插队。一旦将一个执行时间较短的作业放在一个执行时间较长的作业后面执行,就会让该作业等待时间过长,增加了作业的等待时间[56];若作业不在队首,即便有足够的空闲资源满足作业,作业也无法运行,系统资源得不到充分利用。短作业优先SJF(Shortest Job First)对FCFS进行了改进,按照作业的大小顺序进行排序,越短的作业越先运行,从而缩短了作业平均周转时间。

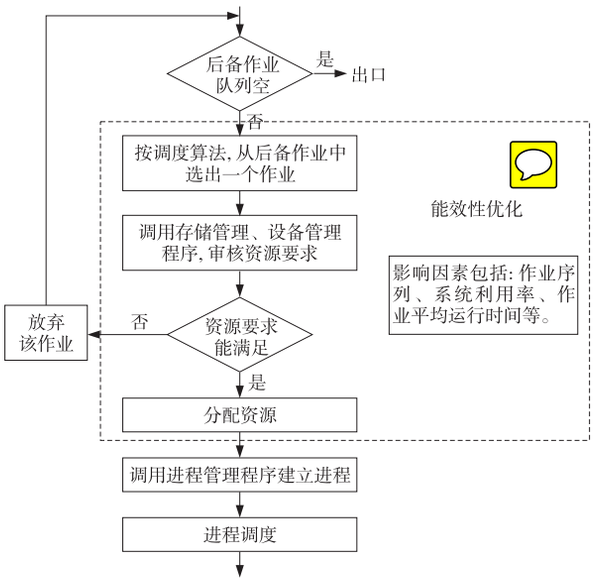


Figure 1 Job scheduling process and its energy efficiency optimization

图 1 作业调度流程及其能效性优化

同样地,Maheswaran 等人[57]提出的 MinET 就是利用类似的作业长度进行排序。由于 FCFS 方式只考虑每个作业的等待时间而未考虑执行时间的长短,而 SJF 方式只考虑执行时间而未考虑等待时间的长短,因此 SJF 出现了一些变形,如最短剩余时间优先 SRT(Shortest Remaining Time)和最高响应比优先 HRN(Highest Response-ratio Next),但仍然无法避免系统开销的增加。

### 3.2 回填策略

回填(BackFill)策略[56,58]允许作业插队,原则是优先作业的开始执行时间不可被延迟。策略根据用户提供的估计作业执行时间,为作业预留资源,在不违反原则的前提下,选择非队首作业尝试运行。回填的有效性取决于系统对执行时间估计的准确性,使得被回填的作业优于其他作业,从而提升整体作业集的性能。如果实际运行作业的运行时间长于预期值,系统将终止作业。尽管如此,即使存在不准确的估计,BackFill 也减少了系统的碎片,增加了系统利用率,并最小化作业运行时间,提高了系统整体性能。美国 Argonne 国家实验室在 IBM SP2 超级计算机上使用的可扩展 Argonne 调度系统 EASY(the Extensible Argonne Scheduling sYstem)[56],就是运用了回填技术。与回填策略类似的首次适配 FF(First-Fit)是在作业队列中查找能够与第一个可用空闲资源相匹配的作业,从而减少空闲结点数量,提高资源利用率和缩短作业平均等待时间,但可能导致作业饿死。

回填策略比先来先服务策略更能保证系统不会有作业饿死,而且回填策略更利于提高系统利用率和降低作业平均等待时间。大多数情况下,回填策略比先来先服务策略更能提高系统性能,而较首次适配策略性能差距较小,且用户的不准确估计并不会对回填策略的表现造成大的影响,因此成为了大多数调度系统的选择。

## 4 作业调度能效性分析

在作业调度过程中,尤其对于并行计算系统,作业调度策略往往造成运行速度快、执行时间短的结点等待运行速度慢、执行时间长的结点[25]。不同结点之间的负载不平衡不仅带来不必要的能耗,还会增加整个应用程序的执行时间,导致空闲结点能量白白浪费掉。作业调度系统节能的目标是在满足性能要求的前提下,所有任务消耗的总功耗最低,这是个 NP 难问题[18,48,49,59]。为此,研究者做了大量的实验工作,结合不同的调度策略和能效性技术[60],针对负载功率与负载时长[61],从应用层和系统层对作业调度能效性进行优化。典型的作业调度能效性策略主要考虑 2 个方面:(1)将作业负载整合到最少的计算资源中;(2)增加可转换为低功耗模式的计算资源数量[62]。下面对一些主流的作业调度能效性策略进行介绍,并在表 1 中进行了对比分析。

**Table 1 Analysis of job scheduling energy efficiency**

**表 1 作业调度能效性分析**

| 优化策略 | 优势 | 不足 |
|---|---|---|
| 缩短总繁忙时间 | 缩短作业运行时间,提高资源利用率 | 未考虑异构性对能耗的影响 |
| 调度算法优化 | 优化作业-资源分配 | 对系统参数依赖大,与其它作业调度策略结合性差 |
| 自适应功耗管理策略 | 提高资源利用率,降低系统整体功耗 | 未充分考虑通信对能耗的影响 |
| 功率封顶(预算)技术 | 控制系统整体功耗 | 功率预算精度问题 |
| 热管理技术 | 控制系统整体能耗 | 对设备要求高 |

### 4.1 缩短总繁忙时间

评价一个作业调度系统,最普遍的标准就是衡量其负载运行时间和系统资源利用率,而降低系统能耗也是从这 2 个方面入手。为了节省能耗,大多数以往的研究选择降低结点频率或者直接关闭一些设备的方式,而忽略了资源利用率这一方面,这往往会造成计算资源的闲置与浪费。目前已经有

许多研究致力于优化作业调度策略,使得服务器总繁忙时间(即如果服务器某一时刻至少有一个作业正在运行,那么此服务器在这一时刻是繁忙的,所有繁忙的时间总和,就是总繁忙时间)最小化,或者是在给定的繁忙时间预算约束下最大化系统资源利用率。

Shalom 等人[63]首先提出线上调度总繁忙时间最小化的目标,这一点比较符合当前用户缩短运行时间,以节省计算费用的需求(但没有考虑异构系统对能耗的不同影响)。蔡立军等人[61]立足降低服务器运行时间,分析服务器处理作业请求的繁忙时间与能耗之间的关系,提出一种基于繁忙时间的并行调度能耗优化算法——BTEOA(Busy Time Energy Optimize Algorithm)。在保持作业请求原有的排队序列基础上,调度作业请求到能使所有服务器总繁忙时间局部最优的服务器上执行,以降低系统能耗,同时保证原有作业调度性能不受影响。但是,由于服务器系统能耗是由繁忙状态和空闲状态能耗组成,该方法只考虑服务器在处理作业请求时处于繁忙状态的时间与能耗,将功率消耗与服务 CPU 利用率之间的关系单纯地设为线性关系,而忽略了空闲结点的能耗问题。

### 4.2 调度算法优化

作业调度能效性优化是 NP 难问题,除了单纯地使用 FCFS、回填或者其它方法外,还可以运用算法优化、路径选择等方法进行作业-资源分配。祝明发等人[64]将用户提交的作业划分为具有依赖关系的多个独立任务,形成标准有向无环图 DAG(Directed Acyclic Graph)模型[50,65],再根据 DAG 获知任务的前驱和后继信息,为每个任务设置 1 个信息表和 1 个通信队列,利用作业间通信消耗选择分配给作业的资源。

Shalom 等人[63]利用贪婪算法进行调度,首先根据设定的范围划分参数生成多个初始作业区,然后按照作业请求执行时间长短,将作业请求分装在不同的作业区,最后将作业区调度到满足条件的服务器上执行,以达到缩短服务器总繁忙时间的效果。Khandekar 等人[66]提出最长处理时间优先结合最早开始时间作业优先 MFFDE(ModiFied First Decreasing Earliest)的调度方法,以最小化服务器总繁忙时间。Tian 等人[67]提出动态二分优先 BFF(Bipartition First Fit)的调度方法,将作业请求按照结束时间分为 2 个执行窗口依次执行,然后按照服务器首次适配策略 FF 进行匹配调度,来最小化服务器总繁忙时间,从而降低系统能耗。

这些现有的方法虽然能够在一定程度上降低能耗,但是也存在一些不足之处,比如,经常对 CPU 调频调压不但会影响服务器的计算性能,同时也影响硬件的电气性能;贪婪算法需要每个作业区只对应 1 个服务器,在一般的情形中作业区划分参数很难确定,任何不当的参数都会引起资源利用不充分的情况发生;缩短服务器总繁忙时间,多数都对作业请求与服务器匹配只是单纯地应用了 FF 的服务器匹配策略,随机性大。此外,在作业请求调度过程中忽视了其对作业调度性能的影响,对作业请求排队模型进行了变更操作,牺牲了部分作业调度性能,无法与其他有调度性能优势的作业调度算法结合使用。

### 4.3 自适应功耗管理策略

目前,许多研究致力于对高性能计算系统级作业功耗监控分析,并依此进行最优化功耗调度策略的探索,总结出符合实际应用情况的功耗分析和管理算法[68]。

王洁等人[69]提出的自适应功耗管理策略就是基于作业的周期性和连续性分析,利用遗传算法的自适应功耗调度策略作为作业调度算法,通过任务之间的相关性和出现概率对负载变化进行预测,根据通信需求和预测的负载确定任务在不同结点上的运行时间和功耗,最小化两者的乘积(即能耗),达到提高资源利用率和降低系统整体功耗的目的。Wallace 等人[41]发现高性能计算作业之间显著不同的功耗分布,并以此通过动态学习技术跟踪观察、分析、评估系统和用户作业进入、执行、退出时的能耗数据,以"数据驱动"的方式进行作业调度,以节省能耗,再利用基于窗口的调度方式提高系统资源利用率,满足用户给定的任意功耗限制,达到了节省系统能耗的效果。

由于作业具有一定的连续性和周期性,不仅仅是结点负载,通信开销也与应用相关。随着资源利用率增加,其计算负担和通信量也会随之增加,能耗也会增加。通过有效地调度分配资源,以减少计算负载和处理时间,可以节省总体电量,尤其是运用 MPI 的并行程序,对于分布式存储系统,通信量对于总能耗的影响是不可忽视的[54]。

### 4.4 功率封顶(预算)技术

Ranganathan 等人[70]的研究表明,大规模集群计算资源利用率大多数时间是维持在低水平的,实际使用的最大功率与其理论峰值之间可能相差 40%,而且极少发生异常情况。这就为整个高性能

计算系统设置一个功耗阈值提供了可能性,系统不仅能正常运行,同时整体功耗得到了控制。目前功率封顶(预算)技术已经在许多数据中心上得以实现,Etinski 等人[5]提出的并行作业调度策略就是利用 DVFS 调整结点功耗,确保整个系统功耗维持在一个给定的功率限度下,Wallace 等人[41]也是在用户给定的功率预算情况下利用数据驱动方式对 IBM-Q 上的作业进行调度,以控制高性能计算系统功耗的。

### 4.5 热管理技术

由于内核温度每升高 10℃,系统错误率就增加 1 倍[71],较高的温度对系统的可靠性有很大的影响,同时也增加了冷却成本。在作业调度过程中应用热管理技术,与 DPM、功率封顶技术类似,但热管理更偏向于硬件层。研究者可以利用热量传感器和热量调节系统,根据预定的温度阈值调整系统工作负荷和平衡负载,从而提高并行设备的可靠性,并降低应用程序的整体运行时间[41,72]。Sarood 等人[72]用实验证明了控制内核温度与作业调度过程中各结点平衡负载的可能性,设备可靠性不仅提高了 2.3 倍,程序执行时间也缩短了 12%。当给定结点温度超过阈值时,系统工作量就会减少。热管理技术的缺点是响应不及时,存在过热、过冷的风险[73],而且对于数以千计的系统组件,对其温度进行全面实时的监控就需要更多的监控器。因此,该技术很少运用在作业调度中,更多的是用于整体系统热量的统计。

## 5 面临的技术挑战和未来发展方向

随着高性能计算技术的飞速发展,计算环境越来越复杂,如何充分利用有限的计算资源,满足用户的应用需求,同时达到最佳能效性,始终是作业调度策略的最基本、也是亟需解决的问题。本节主要从计算认知性、设备异构性、调度公平性等方面分析作业调度能效性所面临的技术挑战和未来发展。

### 5.1 认知计算

为了描述让计算机系统能够像人的大脑一样学习、思考,并做出正确决策的现象,人们提出“认知计算”的概念。它源自模拟人脑的计算机系统的人工智能,试图解决生物系统中的不精确、不确定和部分真实的问题,以实现感知、记忆、学习、语言、思维和问题解决等过程,不同于传统定量、着重于精度和序列的计算技术。随着科技发展以及大数据时代的到来,如何实现类似人脑的认知与判断,发现新的关联和模式,从而做出正确的决定,显得尤为重要。Gaussier 等人[74]利用经典的机器学习方法从大量的实测数据中预测出作业运行时间,优化了 EASY 回填算法,使作业调度性能提升了 28%。显然,大数据给认知计算领域带来新的机遇和挑战的同时,也给超大规模的作业调度技术及其能效性提出了新的要求——以非常灵活的方式进行作业调度,并以交互的方式指导计算,以支持人类的认知过程[52]。

### 5.2 设备异构性

如今的高性能计算的工作负载和基础设备日益复杂,即使是在单个结点中,核和 GPU 的数量也在不断变大,系统的异构性导致功耗模式、功率状态、功率管理机制、计算能力和通信模式的不同,造成的系统能耗也会截然不同[21],对如此复杂的大规模系统作业进行调度是当前最主要的挑战之一,迫切需要更加先进的调度方法。尤其表现在作业调度的模拟器及其系统性能评估基准的创新方面。Zong 等人[7]将异构系统分为 4 种类型,并将作业调度区分组建和分配 2 个阶段,预估结点上运行的作业能耗,进而决定最佳的分配策略,以最小化系统总能耗,达到较好的节能效果。虽然最新的研究设计出调度系统模拟器(ScSF[75])及基准(DJSB[76]),但也仅局限于对大规模作业/应用程序进行资源分配时的性能测试,对于作业调度能效性的研究仍然有待进一步深入。

### 5.3 调度公平性

用户提交的作业一直以来都有着不同计算需求:有的用户对于精确度要求高,只要求作业能在一定期限内完成;有的用户对作业计算时间要求高,越早完成越好;有的用户既不要求高速度,也不要求高精度;有的则相反。不管系统结构如何不一样,计算规模有多大,调度的公平性始终是设计人员必须关注的一个重要方面,那么,如何在保证调度公平性的前提下,实现最佳的能效性,也是研究人员所应该追求的目标。进一步地,大多数批处理调度程序都是基于作业队列的,其中每个作业在其到达前就已经计划好其调度序列。在云数据时代,对大规模数据进行作业调度时,能否及时对如此庞大的作业数量进行规划并提交给用户,同时确保作业-资源分配的公平性,成为应用程序能否达到最佳性能的关键因素。

## 6 结束语

高性能计算过程中,大量的能耗已成为技术发展的瓶颈。作业调度系统作为高性能计算机中的核心组件,对于提高系统性能,合理利用有限的计算资源有着关键作用,其能效性也决定着性能所能提升的空间。本文首先介绍传统能效性技术和作业调度策略,并就最新的作业调度技术优化及其能效性进行了论述,最后是作业调度能效性方面面临的挑战及未来走向。目前作业调度策略在能效性方面的研究仍有待于进一步深入。

**参考文献:**


[1] Agerwala T. Challenges on the road to exascale computing [C]// Proc of the 22nd Annual International Conference on Supercomputing (ICS'08),2008:2.

[2] Lucas R,Ang J,Bergman K,et al. Top ten exascale research challenges: DOE ASCAC Subcommittee Report [R]. New York:USDOE Office of Science,2014:1-2.

[3] Wu C. Making a case for efficient supercomputing [J]. ACM Queue,2003,1(7):54-64.

[4] Ge R,Feng X,Cameron K W. Improvement of power-performance efficiency for high-end computing [C]// Proc of the 19th IEEE International Parallel and Distributed Processing Symposium,2005:1-8.

[5] Etinski M,Corbalan J,Labarta,et al. Parallel job scheduling for power constrained HPC systems [J]. Parallel Computing,2012,38(12):615-630.

[6] Srinivasan S,Jha N K. Safety and reliability driven task allocation in distributed systems [J]. IEEE Transactions on Parallel & Distributed Systems,1999,10(3):238-251.

[7] Zong Z, Qin X, Ruan X,et al. Energy-efficient scheduling for parallel applications running on heterogeneous clusters[C]// Proc of the 36th International Conference on Parallel Processing(ICPP'07), 2007:19-26.

[8] Liao Xiang-ke,Pang Zheng-bin,Wang Ke-fei,et al. High performance interconnect network for Tianhe system[J]. Journal of Computer Science and Technology,2015,30(2):259-272.

[9] Pang Zheng-bin,Xie Min,Zhang Jun,et al. The TH express high performance interconnect networks [J]. Frontiers of Computer Science,2014,8(3):357-366.

[10] Valentini G L,Lassonde W,Khan S U,et al. An overview of energy efficiency techniques in cluster computing systems [J]. Cluster Computing,2013,16(1):3-15.

[11] Li K. Energy efficient scheduling of parallel tasks on multiprocessor computers [J]. The Journal of Supercomputing,2012,60(2):223-247.

[12] Masanet E R,Brown R E,Shehabi A,et al. Estimating the energy use and efficiency potential of U. S. data centers [J]. Proceeding of the IEEE,2011,99(8):1440-1453.

[13] Shuja J, Madani S A,Bilal K,et al. Energy-efficient data centers [J]. Computing,2012,94(12):973-994.

[14] Koomey J G. Growth in data center electricity use 2005 to 2010 [M]. Oakland:Analytics Press,2011.

[15] Koomey J G. Worldwide electricity used in data centers [J]. Environmental Research Letters, 2008 (3): Article ID 034008.

[16] Hikita J,Hirano A,Nakashima H. Saving 200kW and $200 K/year by power-aware job/machine scheduling [C]// Proc of 2008 IEEE International Symposium on Parallel and Distributed Processing(IPDPS 2008),2008:1-8.

[17] Ge R,Feng X,Cameron K W. Performance-constrained distributed DVS scheduling for scientific applications on power-aware clusters[C]// Proc of ACM/IEEE Conference on Supercomputing (SC'05),2005:34-44.

[18] Cao Z,Watson L T,Cameron K W,et al. A power aware study for VTDIRECT95 using DVFS [C]// Proc of the 2009 Spring Simulation Multiconference,2009:531-536.

[19] Weiser M, Welch B, Demers A, et al. Scheduling for reduced CPU energy [C]// Proc of the 1st USENIX Conference on Operating Systems Design and Implementation (OSDI'94),1994:13-23.

[20] Gruian F,Kuchcinski K. LEneS: Task scheduling for low-energy systems using variable supply voltage processors [C]// Proc of the ASP-DAC,2001:449-455.

[21] Horvath T,Abdelzaher T,Skadron K,et al. Dynamic voltage scaling in multitier web servers with end-to-end delay control [J]. IEEE Transactions on Computers,2007,56(4):444-458.

[22] Zhong X L, Xu C Z. System-wide energy minimization for real-time tasks: Lower bound and approximation [J]. ACM Transactions on Embedded Computing System, 2008,7(3):1-24.

[23] Bilal K,Khan S U,Madani S A,et al. A survey on green communications using adaptive link rate [J]. Cluster Computing,2013,16(3):575-589.

[24] Pinheiro E,Bianchini R,Carrera E V,et al. Load balancing and unbalancing for power and performance in cluster-based systems[C]// Proc of the Workshop on Compilers & Operating Systems for Low Power,2001:1-8.

[25] Liu Y,Zhu H. A survey of the research on power management techniques for high-performance systems[J]. Software Practice & Experience,2010,40(11):943-964.

[26] Intel Corporation. ACPI: Advanced configuration and power interface[EB/OL]. [2019-02-26]. http://www.acpi.info/.

[27] Intel Corporation. Intel ® 64 and IA-32 architectures software developer's manual[EB/OL]. [2019-02-26]. https://software.intel.com/en-us/articles/intel-sdm.

[28] Intel Corporation. Difference between deep and deeper sleep states for processors [EB/OL]. [2019-02-26]. https://www.intel.com/content/www/us/en/support/articles/000006619/processors/intel-core-processors.html.

[29] Janzen J. Calculating memory system power for DDR SDRAM[J]. Micro Designline,2001,10(2):1-12.

[30] Pandey V,Jiang W,Zhou Y,et al. DMA-aware memory energy management[C]// Proc of the 12th International Symposium on High-Performance Computer Architecture(HPCA'06),2006:133-144.

[31] Colarelli D, Grunwald D. Massive arrays of idle disks for storage archives[C]// Proc of the 2002 ACM/IEEE Conference on Supercomputing(SC'02),2002:1-11.

[32] Pinheiro E,Bianchini R,Dubnicki C. Exploiting redundancy to conserve energy in storage systems[J]. ACM SIGMETRICS Performance Evaluation Review,2006,34(1):15-26.

[33] Chen G,He W,Liu J,et al. Energy-aware server provisioning and load dispatching for connection-intensive internet services[C]// Proc of the 5th USENIX Symposium on Networked Systems Design and Implementation(NSDI'08),2008:337-350.

[34] Liu J, Zhao F, Liu X, et al. Challenges towards elastic power management in internet data centers[C]// Proc of the 2nd International Workshop on Cyber-Physical Systems(WCPS'09),2009:65-72.

[35] Kliazovich D,Bouvry P,Khan S U. DENS:Data center energy-efficient network-aware scheduling[J]. Cluster Computing,2013,16(1):65-75.

[36] Chen C R,Chen J,Chang E C. Reducing static energy in supercomputer interconnection networks using topology-aware partitioning [J]. IEEE Transactions on Computers,2016,65(8):2588-2602.

[37] Benini L,Bogliolo A,Micheli G D. A survey of design techniques for system-level dynamic power management[J]. Hardware/Software Co-Design,2000,8(3):299-316.

[38] Meisner D,Wenisch T F. DreamWeaver:Architectural support for deep sleep[J]. ACM SIGARCH Computer Architecture News,2012,40(1):313-324.

[39] Louis R. Dryad:Distributed data-parallel programs from sequential building blocks[J]. ACM SIGOPS Operating Systems Review,2007,41(3):59-72.

[40] Nedevschi S,Popa L,Iannaccone G,et al. Reducing network energy consumption via sleeping and rate-adaptation[C]// Proc of USENIX Symposium on Networked Systems Design & Implementation(NSDI'08),2008:323-336.

[41] Wallace S, Yang X, Vishwanath V,et al. A data driven scheduling approach for power management on HPC systems in High Performance Computing[C]// Proc of the International Conference for High Performance Computing, Networking,Storage and Analysis(SC'16),2016:56:1-56.

[42] Graham S L,Snir M,Patterson C A. Getting up to speed:The future of supercomputing [M]. Washington:Committee on the Future of Supercomputing,National Research Council,National Academies Press,2005.

[43] Chen Y,Das A,Qin W,et al. Managing server energy and operational costs in hosting centers [C]// Proc of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems,2005:303-314.

[44] Horvath T,Skadron K. Multi-mode energy management for multi-tier server clusters[C]// Proc of the 17th International Conference on Parallel Architectures and Compilation Techniques(PACT'08),2008:270-279.

[45] Wu F,Chen J,Chen Z,et al. A holistic energy-efficient approach for a processor-memory system[J]. Tsinghua Science and Technology,2019,24(4):468-483.

[46] Komoda T,Hayashi S,Nakada T,et al. Power capping of CPU-GPU heterogeneous systems through coordinating DVFS and task mapping [C]// Proc of the 31st International Conference on Computer Design(ICCD 2013),2013:349-356.

[47] Mei J,Li K L,Li K Q. Energy-aware task scheduling in heterogeneous computing environments [J]. Cluster Computing,2014,17(2):537-550.

[48] Garey M R,Johnson D S. Computers and intractability: A guide to the theory of NP-completeness [M]. New York: W. H. Freeman & Co,1979.

[49] Ullman J D. NP-complete scheduling problems [J]. Journal of Computer and Systems Sciences,1975,10(3):384-393.

[50] Kwok Y K,Ahmad I. Benchmarking the task graph scheduling algorithms [C]// Proc of the 1st International Parallel Processing Symposium,1998:531-537.

[51] Wu F,Juan C,Dong Y,et al. Improve energy efficiency by processor overclocking and memory frequency scaling [C]// Proc of the 20th International Conference on High Performance Computing and Communications(HPCC'18),2018:960-967.

[52] Feitelson D G,Rudolph L,Schwiegelhohn U,et al. Theory and practice in parallel job scheduling[J]. Lecture Notes in Computer Science,1997,1291(13):1-34.

[53] Majumdar S,Eager D L,Bunt R B. Scheduling in multiprogrammed parallel systems [C]// Proc of SIGMETRICS Conference on Measurement and Modeling of Computer Systems,1988:104-113.

[54] Cime W,Desai N. Job scheduling strategies for parallel processing [J]. Computer Science,2014,2862(4):128-152.

[55] Celik B,Suryanarayanan S,Maciejewski A A,et al. A comparison of three parallel processing methods for a resource allocation problem in the smart grid[C]// Proc of 2017 North American Power Symposium (NAPS),2017:1-7.

[56] Lifka D. The ANL/IBM SP scheduling system[C]// Proc of the Workshop on Job Scheduling Strategies for Parallel Processing(IPPS'95),1995:187-191.

[57] Maheswaran M,Ali S,Siegil H J,et al. Dynamic mapping of a class of independent tasks onto heterogeneous computing systems[J]. Journal of Parallel & Distributed Computing,1999,59(2):107-131.

[58] Mu'alem A W, Feitelson D G. Utilization, predictability, workloads, and user runtime estimates in scheduling the IBM SP2 with backfilling[J]. Parallel & Distributed Systems,2001,12(6):529-543.

[59] Dong Jing. Design and research of low power resource management in high performance parallel computing system [D]. Changsha:National University of Defense Technology, 2009. (in Chinese)

[60] Chandio A A,Bllal K,Tziritas N,et al. A comparative study on resource allocation and energy efficient job scheduling strategies in large-scale parallel computing systems[J]. Cluster Computing,2014,17(4):1349-1367.

[61] Cai Li-jun,Pan Jiang-bo,Chen Lei,et al. A parallel scheduling energy consumption optimization algorithm based on busy hours[J]. Computer Engineering & Science,2017,39(1):42-48. (in Chinese)

[62] Li B,Li J,Huai J,et al. EnaCloud:An energy-saving application live placement approach for cloud computing environments[C]// Proc of 2009 IEEE International Conference on Cloud Computing,2009:17-24.

[63] Shalom M, Voloshin A, Wong P W H,et al. Online optimization of busy time on parallel machines[C]// Proc of the 9th Annual Conference on Theory and Applications of Models of Computation,2012:448-460.

[64] Zhu Ming-fa,Pang Yu,Liang Ai-hua,et al. A job scheduling method for optimizing the energy consumption of multi-task communication:China, 201110333204. 3 [P]. 2011-10-28. (in Chinese)

[65] Ahmad I. Editorial:Resource management of parallel and distributed systems with static scheduling:Challenges,solutions and new problems[J]. Concurrency & Computation Practice & Experience,2010,7(5):339-347.

[66] Khandekar R, Schieber B, Shachnai H, et al. Minimizing busy time in multiple machine real-time scheduling[C]// Proc of IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science,2010:169-180.

[67] Tian W, Xiong Q, Cao J. An online parallel scheduling method with application to energy-efficiency in cloud computing[J]. Journal of Supercomputing, 2013, 66(3): 1773-1790.

[68] Klusácek D,Parák B. Analysis of mixed workloads from shared cloud infrastructure[C]// Proc of Workshop on Job Scheduling Strategies for Parallel Processing,2017:25-42.

[69] Wang Jie, Zeng Yu. Research on job scheduling strategy of high performance computer based on adaptive power management [J]. Computer Science,2012,39(10):313-317. (in Chinese)

[70] Ranganathan P, Leech P, Irwin D, et al. Ensemble-level power management for dense blade servers[C]// Proc of the 33rd International Symposium on Computer Architecture, 2006:66-77.

[71] Wu C. The importance of being low power in high performance computing[J]. Cyberinfrastructure Technology Watch Quarterly (CTWatch Quarterly),2005,1(3):12-20.

[72] Sarood O,Meneses E,Kale L V. A 'cool' way of improving the reliability of HPC machines[C]// Proc of the International Conference for High Performance Computing, Networking,Storage and Analysis,2013:Article No. 58.

[73] Lee E K, Kulkarni I, Pompili D, et al. Proactive thermal management in green datacenters[J]. Journal of Supercomputing,2012,60(2):165-195.

[74] Gaussier E,Glesser D,Reis V,et al. Improving backfilling by using machine learning to predict running time[C]// Proc of the International Conference for High Performance Computing,Networking,Storage and Analysis,2015:1-64.

[75] Rodrigo G P,Elmroth E,Östberg P O,et al. ScSF:A scheduling simulation framework[C]// Proc of Workshop on Job Scheduling Strategies for Parallel Processing,2017:152-173.

[76] Lopez V,Jokanovic A,Amico M D,et al. DJSB:Dynamic job scheduling benchmark[C]// Proc of Workshop on Job Scheduling Strategies for Parallel Processing, 2018: 174-188.

## 附中文参考文献:

[59] 董晶.高性能并行计算系统中低功耗资源管理的设计与研究[D].长沙:国防科学技术大学,2009.

[61] 蔡立军,潘江波,陈磊,等.一种基于繁忙时间的并行调度能耗优化算法[J].计算机工程与科学,2017,39(1):42-48.

[64] 祝明发,庞瑜,梁爱华,等.一种优化多任务间通信能耗的作业调度方法:中国,201110333204.3[P].2011-10-28.

[69] 王洁,曾宇.基于自适应功耗管理的高性能计算机作业调度策略的研究[J].计算机科学,2012,39(10):313-317.

## 作者简介:

**郑文旭**(1988-),男,湖南邵东人,硕士生,研究方向为系统软件。**E-mail**:zhengwen5241@qq.com

**ZHENG Wen-xu**, born in 1988, MS candidate, his research interest includes system software.

**潘晓东**(1989-),男,河南驻马店人,硕士生,研究方向为高性能计算。**E-mail**:sheldon.pan@hotmail.com

**PAN Xiao-dong**, born in 1989, MS candidate, his research interest includes high performance computing.

**马迪**(1988-),男,四川广元人,硕士,工程师,研究方向为高性能计算。**E-mail**:535907803@qq.com

**MA Di**, born in 1988, MS, engineer, his research interest includes high performance computing.

**汪浩**(1990-),男,安徽铜陵人,硕士生,研究方向为高性能计算。**E-mail**:wanghao18d@nudt.edu.cn

**WANG Hao**, born in 1990, MS candidate, his research interest includes high performance computing.