虚拟化云计算平台的能耗管理

叶可江 吴朝晖 姜晓红 何钦铭
(浙江大学计算机科学与技术学院 杭州 310027)

摘 要 数据中心的高能耗是一个亟待解决的问题.近年来,虚拟化技术和云计算模式快速发展起来,因其具有资源利用率高、管理灵活、可扩展性好等优点,未来的数据中心将广泛采用虚拟化技术和云计算技术.将传统的能耗管理技术与虚拟化技术相结合,为云计算数据中心的能耗管理问题提供了新的解决思路,是一个重要的研究方向.文中从能耗测量、能耗建模、能耗管理实现机制、能耗管理优化算法4个方面对虚拟化云计算平台能耗管理的最新研究成果进行了介绍.论文分析了虚拟化云计算平台面临的操作管理和能耗管理两方面的问题,指出了虚拟化云计算平台能耗监控与测量的难点;介绍了能耗监测步骤及能耗轮廓分析方法;提出了虚拟机系统的整体能耗模型及服务器整合和在线迁移两种关键技术本身的能耗模型;从虚拟化层和云平台层两个层次总结了目前能耗管理机制方面取得的进展;并对能耗管理算法进行分类、比较.最后对全文进行总结,提出了未来十个值得进一步研究的方向.

关键词 虚拟化;云计算;能耗管理;绿色计算;在线迁移;服务器整合
中图法分类号 TP393 **DOI号**: 10.3724/SP.J.1016.2012.01262

Power Management of Virtualized Cloud Computing Platform

YE Ke-Jiang WU Zhao-Hui JIANG Xiao-Hong HE Qin-Ming
(*College of Computer Science and Technology, Zhejiang University, Hangzhou* 310027)

Abstract The high energy consumption of data center is a serious problem. Recently, with the rapid development of virtualization technology and the emergence of cloud computing paradigm, a large number of future-generation data center will use virtualization technology and cloud computing technology, due to the benefits of high resource utilization, flexible management, and dynamic scalability. The integration of traditional power management technology and virtualization technology provides new solutions to solve the power management issue of cloud data center, which is an important research direction. This paper focuses on the virtualized cloud computing platform, surveys the latest research results from the perspectives of power measurement, power modeling, implementation mechanisms, and optimization algorithms. We investigate the challenges of operation and power management in virtualized cloud platform, and try to find out the difficulties of power monitoring and measurement technology. We introduce the steps of power measurement and power profiling technology, and try to build the power model for the whole virtual machine system, also the power model for the technique of server consolidation and live migration. We summarize the advances on power management mechanisms from both virtualization level and cloud level, and classify and compare the existing power management algorithms. Finally, we summarize the contents of this paper and propose ten possible research directions in the future.

Keywords virtualization; cloud computing; power management; green computing; live migration; server consolidation

收稿日期:2011-08-27;最终修改稿收到日期:2012-04-16.本课题得到国家"八六三"高技术研究发展计划重大项目基金(2011AA01A207)、国家自然科学基金项目(11071215)和教育部-英特尔专项基金(MOE-INTEL-11-06)资助.**叶可江**,男,1986年生,博士研究生,主要研究方向为虚拟化与云计算、性能评估与建模.E-mail:yekejiang@zju.edu.cn.**吴朝晖**,男,1966年生,博士,教授,主要研究领域为服务科学与网格计算、嵌入式普适计算等.**姜晓红**(通信作者),女,1966年生,博士,副教授,主要研究方向为计算机体系结构、分布式系统、云计算等.E-mail:jiangxh@zju.edu.cn.**何钦铭**,男,1965年生,博士,教授,主要研究领域为虚拟化技术、机器学习等.

1 引　言

近年来，数据中心的高能耗逐渐成为一个突出的问题，尤其是随着云计算时代的到来，更多的计算资源和存储资源集中在云端，给能耗的高效管理带来更大的挑战[1-2]. 据统计，2006 年美国 6000 个左右的数据中心，消耗了大约 610 亿千瓦时的电能，总值高达 45 亿美元，超过了当年美国所有彩色电视机的总能耗[3]. 来自美国能源部的数据表明，数据中心的能耗占全美所有能耗的 1.5%，并且对电能的需求仍在以每年 12%的速度增长. 如按这种速度增长，到 2011 年，数据中心会消耗 1000 亿千瓦时的电能，每年花费约 74 亿美元[3]. 此外，IDC(International Data Corporation)市场研究公司对全球所有企业电能花费的评估结果表明，每年全球的企业大概要花费 400 亿美元在能耗上[4]. 数据中心的高能耗问题不仅造成电能的浪费、系统运行的不稳定，同时也对环境造成不良影响. 美国联邦机构已经指出高能耗问题将对空气质量、国家安全、气候变化、电网可靠性等方面造成严重影响.

高能耗问题主要来源于两个方面，一个是处理器层次的能耗，另一个是数据中心层次的能耗. 随着处理器制造工艺的不断进步，今天的 Intel Itanium2 处理器的晶体管数量已达到 10 亿个[5]. 处理器在获得高运行速度的同时，带来了高能耗问题. 尽管一些硬件上的优化技术在一定程度上提升了能量的使用效率，但是处理器的能耗受应用程序使用模式的影响，过高的负载和过低的利用率都会导致高能耗和电能低效使用的问题. 在数据中心层次，随着数据中心规模的不断增长，数据中心出现了两难的境况. 一方面由于物理服务器数量不断增多和处理能力不断增强，带来了更多的能量消耗，另一方面每个服务器过低的利用率又造成了巨大的电能浪费. IBM 的 Bohrer 等人[6]曾对真实的若干典型 Web 服务器负载(体育、电子商务、财经、互联网代理集群)进行研究，发现数据中心服务器的平均利用率在 11%～50%之间. Barroso 等人[7]对超过 5000 台服务器长达 6 个月的运行情况进行统计分析，也得出类似的结论. 高能耗带来的另一个问题是配套的冷却设施开销极大. 据统计，计算资源消耗的每 1 瓦电能，就需要额外的 0.5～1 瓦特进行冷却[8]. 另外，也有统计数据表明，Google 数据中心的能耗有一半是花在了冷却上[3]. 因此，迫切需要开发新的技术来解决数据中心的高能耗问题，包括重新考虑硬件、软件、算法的设计，特别是必须把能耗作为数据中心设计和管理的一个重要参数进行考虑，而不仅仅是考虑数据中心的性能参数.

针对数据中心的高能耗问题，相关政府机构、社会团体、学术组织已经开始积极关注. 2007 年，绿色网格组织(Green Grid)①成立，它的目标就是要降低数据中心和商业计算系统的能耗. 为了达到这个目标，绿色网格组织联合了公司、政府部门、工业界团体，共同研究并试图提供最佳的数据中心节能实践方法、评估方法和技术. 同年，Green500 组织(Green500 List)②成立，该组织每年会发布 2 个报告(分别是在当年的 6 月和 12 月)，以 MegaFlops/W(即每瓦特电力所能完成的每秒百万次浮点计算操作)为指标，对全球范围内最快的 500 台超级计算机的节能效率进行排序，作为对 TOP500 排名的补充. 2008 年，国际气候组织(The Climate Group)③和全球电子可持续发展倡议组织(GeSI)④联合发布了题为《SMART2020：实现信息时代的低碳经济》的报告. 该报告表明，到 2020 年，信息通信技术(ICT)共计能减少近 78 亿吨 CO_2 的排放，相当于 2020 年基准情景下排放总量的 15%. IEEE 也正在制定一项能被国际社会接受的新标准 IEEE P1595⑤，该标准旨在检测、评估和量化由可再生能源和节能措施所带来的温室气体排放的改善效果. Google 也专门成立了一个能量子公司 Google Energy 来减少本公司的能耗，同时也制造和销售清洁能源. 此外，SPEC(Standard Performance Evaluation Corporation)⑥、TPC(The Transaction Processing Performance Council)⑦等国际性能评估标准化组织也都在致力于能耗的标准化评价和优化问题.

当前，虚拟化技术的快速发展给数据中心的能耗管理问题提供了新的解决思路[9]. 尤其是当云计算成为未来数据中心发展的主要方向时，因其拥有服务器整合[10]、在线迁移[11]、隔离性[12]、高可用性[13]、灵活部署[14]、低管理开销等诸多方面的优点，虚拟化技术面临更大的发展空间. 加州大学伯克利

① http://www.thegreengrid.org/
② http://www.green500.org/
③ http://www.theclimategroup.org/
④ http://www.gesi.org/
⑤ http://grouper.ieee.org/groups/1595/
⑥ http://www.spec.org/power_ssj2008/
⑦ http://www.tpc.org/tpc_energy/

分校的 Patterson 教授曾提出“数据中心是一个计算机”的著名论断[15]. 云计算数据中心就是这样一个计算机,它通过网络,以付费即用的方式,为全世界的用户提供基于效用的信息服务. 云计算数据中心需要管理来自不同用户的各种各样的应用程序,如科学计算、商业领域程序等. 许多云计算服务提供商,包括 Amazon、Google、Microsoft、Yahoo 和 IBM 等,在世界各地部署了各自的数据中心来提供云计算服务. 但是,管理这些云程序需要消耗大量电能并带来很高的操作开销,同时也会对环境造成负面影响. 能耗的持续升高会增加云计算基础设施的总拥有成本(Total Cost of Ownership,TCO),降低投资回报率(Return on Investment,ROI). 因此,需要一种绿色的云计算解决方案[16-18],即在节约能耗的同时降低管理操作的开销. 绿色云计算的目标是在高效处理和使用云基础设施的同时最小化能量消耗. 但是当前的云计算基础设施很少提供支持能量感知的服务,无法在满足服务质量需求的同时最小化能耗开销,以获取最大化投资回报.

虚拟化技术给云计算数据中心的能耗管理带来了很多解决思路. 例如,在数据中心层次,虚拟化技术可以通过服务器整合把多个负载整合到同一个物理机上,关闭空闲的物理机,达到节能目的. 但是也正由于虚拟化层的存在,使得传统的细粒度能耗管理技术(如硬件层和操作系统层的能耗管理技术)不能直接应用到虚拟化环境中,如在虚拟机(Virtual Machine,VM)里,资源是虚拟化出来的,它无法直接获得硬件的能耗状态数据. 这些在虚拟化云平台的能耗管理中遇到的新问题,目前正受到越来越广泛的关注,是一个热门的研究方向[19-21].

通常来讲,一个高效的能耗管理解决方案需要考虑 3 方面的关键元素:(1)丰富的能耗监控与测量方法,及时准确地提供原始数据;(2)精确的能耗建模与分析,预测能量的消耗情况,指明趋势及因果关系;(3)节能机制及优化算法,用来降低能耗,同时满足性能、服务质量(Quality of Service,QoS)或服务等级协议(Service-Level Agreement,SLA)等的要求.

本文从能耗监控与测量、能耗分析与建模、能耗管理实现机制、能耗管理优化算法 4 个方面,对虚拟化云计算平台的能耗管理研究进行系统分析. 论文第 2 节分析虚拟化云计算平台面临的操作管理和能耗管理的挑战,指出虚拟化云计算平台中能耗监控与测量的难点,介绍能耗监测步骤及能耗轮廓分析方法;第 3 节提出虚拟机系统的能耗模型及服务器整合和在线迁移两种节能关键技术的能耗模型;第 4 节从虚拟化层和云平台层两个层次总结目前能耗管理机制方面的最新研究进展;第 5 节对常用的能耗管理算法进行分类、比较;最后对全文进行总结,提出未来十个值得进一步研究的方向.

2 虚拟化云平台的能耗监控与测量

2.1 虚拟化云平台的管理挑战

2.1.1 操作管理挑战

虚拟化技术通过动态资源伸缩的方式降低了云计算基础设施的总拥有成本,增加了负载部署的灵活性. VMware 的研究人员通过对来自真实虚拟化部署场景的数据进行分析,总结出虚拟化场景中出现的常见管理工作流,并评估它们对云计算数据中心资源使用的影响[22].

虚拟化技术提供了很好的管理操作灵活性,例如当虚拟化数据中心需要维护时,只需简单地把虚拟机迁移到另外一台服务器上,而不需要终止应用程序并关闭虚拟机. 但虚拟机迁移也会带来一定的开销,如给数据中心网络增加了额外的通信负载和带宽需求,因此需要采用高性能的网络设备来满足负载快速迁移的需求. 此外,虚拟化技术的引入可能改变数据中心的某些决策,例如为了获得虚拟化负载的最佳性能,需要选购最新的具有硬件虚拟化(如 Intel VT 技术)支持的处理器,遗留的处理器将不再适用. 同时由于运行着数以百计关键任务的虚拟机可能同时运行在同一台物理主机上,为了获得更高的可靠性,需要购买更可靠更昂贵的硬件. 这些问题导致在虚拟化数据中心设计的时候需要考虑系统性能和成本的权衡.

虚拟化环境具体会产生什么样的管理开销?为了回答这个问题,VMware 的研究人员收集了运行 VMware 虚拟化软件的 17 个企业数据中心的真实管理操作的详细轮廓数据[22],如表 1 所示. 虚拟化云平台特有的几种管理操作定义如下:(1)虚拟机重配置. 为虚拟机进行硬件的重配置(如增加一块网卡或者增加一块硬盘);(2)自动在线迁移. 为了达到自动负载均衡,虚拟机需要在主机间动态移动,这个过程中虚拟机始终是活着的;(3)虚拟机开启. 开启虚拟机;(4)虚拟机关闭. 关闭虚拟机;(5)虚拟

机重置. 对虚拟机进行软重置(类似于打开物理主机上的重置开关)；(6) 补丁安装. 包括针对物理主机的补丁安装(如更新 Hypervisor)，或者针对虚拟机的程序安装(如更新客户操作系统，运行最新的安全修补程序)；(7) 创建快照. 对虚拟机的状态进行检查点操作；(8) 快照恢复. 如果发生故障，可以回滚到前一个已知的状态；(9) 快照提交. 把所有的变化写到磁盘，真正执行快照操作，并移除临时快照文件；(10) 虚拟机克隆. 创建一个关闭着的虚拟机的拷贝，这对于快速复制一个新的配置非常有用. 例如当一个新员工加入到一个公司，通过虚拟机克隆，可以把标准的桌面虚拟机镜像快速部署到员工的虚拟计算机上. 当进行虚拟化云计算平台的能耗管理时，需要考虑这些虚拟化特有的管理操作.

表 1　虚拟化云数据中心常见的管理操作[22]

操作类型	不同站点每天平均操作次数	不同站点每天峰值操作次数
虚拟机重配置	2.3	699
自动在线迁移	51.0	3156
虚拟机开启	90.0	1576
虚拟机关闭	35.0	1535
虚拟机重置	4.6	176
补丁安装	5.3	250
创建快照	4.8	56
快照恢复	7.0	101
快照提交	13.0	19
虚拟机克隆	6.0	44

2.1.2　能耗管理挑战

虚拟化技术给数据中心带来新的日常管理操作特征的同时，也给能耗管理操作带来新的挑战[19]. 首先，因为虚拟化平台所管理的虚拟资源和物理资源是相互分离的，因而客户机器观察到的虚拟资源与底层物理资源会不一致，特别是出现迁移的时候. 所以，客户虚拟机如何实现虚拟机程序级的能耗管理策略是一个挑战性问题. 其次，在数据中心中，不一致性会随着平台的频繁更新、故障处理、添加新节点扩容等操作而进一步加重. 这些平台变化的一个自然结果是导致异构性的增加. 但是，出于对虚拟机隔离性以及独立性的需求考虑，虚拟机能耗管理策略要做到能够以不变应对多变. 一般来讲，有两种解决方法：

(1) 利用客户虚拟机的能耗管理策略实现虚拟机的能耗管理操作. 但这种方法存在一些问题，例如不能让虚拟机直接使用硬件能耗管理功能，因为这些资源是虚拟化后的硬件资源，并为多个虚拟机所共享. 此外，直接访问硬件资源也会影响性能隔离性. 例如虚拟机增加自己所在物理核的频率来满足自己虚拟机服务质量需求的时候会影响其它的物理核的运行. 更坏的情况是，有些活动可能是恶意的，如功耗病毒(Power Virus)，会危害其它虚拟机的正常运行.

(2) 利用硬件能耗管理机制实现虚拟机的能耗管理操作. 但这种方法也存在局限性. 硬件支持的能耗管理机制对于虚拟化系统的问题是，这些硬件层次的能耗管理策略会被多个客户虚拟机所共享. 例如，内存 DIMM(Dual-Inline-Memory-Modules)通过同一个总线进行访问，并且共享相同的电压水平，而这些内存可能被分配给多个客户虚拟机. 这意味着，这个部件不能被用来进行特定虚拟机的能耗管理操作，除非所有的客户机都想通过总线调节来降低内存带宽. 但若真的这样做，又相当于没有发挥 DIMM 的优势，可以直接关闭 DIMM 了. 相同的局限性也反映在磁盘上，磁盘包括多个分区，每个分区可能被分配给多个不同的客户机. 只有当所有的分区都能被设置为某个功率状态时，才能通过把磁盘调整到那个功率状态，实现磁盘分区层次的能耗管理. 一个解决方法是采用时间域多路复用，就是根据当前虚拟机的功耗标准设置硬件状态. 但是这样也存在两个问题：① 首先，只有在资源管理状态转换时间比 Hypervisor 的调度时间粒度更小的时候才能采用这种方法. ② 这种方法对多核平台不一定有用. 大量的客户虚拟机在可用的物理核上并发执行，降低了时间域多路复用的可用概率. 因为在多核上，尽管核的频率可以独立调节，但是主板传过来的只有一个电压. 因此工作电压受最高频率核的约束.

综上所述，需要研究一套同时结合“软”、“硬”能耗调节技术的方法来进行虚拟化云计算平台的能耗管理.

2.2　虚拟机能耗测量挑战

2.2.1　虚拟机能耗无法直接测量

在真实的云计算系统中，能耗管理技术的一个重要方面是能耗使用情况的可视性. 基于这些可视信息，可以进行自动的或者人工的能耗管理决策，这就涉及一个重要问题，即虚拟机的能耗测量. 由于现代数据中心对能耗管理的内在需求，目前绝大多数的新服务器都在硬件层提供了内置的能耗测量功能，而旧的服务器也有其它一些解决方案，如使用功耗分布单元(Power Distribution Units，PDU)，通过电源测试系统的能耗. 但是在虚拟化环境中，虚拟机的能耗无法直接从硬件测量.

Kansal 等人[23]提出了一个间接的虚拟机能耗测量机制 Joulemeter，使得虚拟化平台也能像目前硬件上提供给服务器的功耗测量功能一样使用能耗管理机制. 首先跟踪虚拟机使用的每个硬件部件的资源使用情况，然后通过一个资源能耗模型，把资源使用率转换为能耗使用率. 有了资源能耗模型，就可以根据模型从运行时资源使用情况推断出虚拟机的能量消耗. 传统的能耗模型对于运行单个应用程序的物理服务器是有效的，但不适合于云计算平台. 在云计算平台中一个服务器可能被多个不同的虚拟机共享，每个虚拟机运行不同的应用程序. Joulemeter 通过服务器硬件和对 Hypervisor 插桩(Instrumentation)来建立所需的基于真实平台的能耗模型. 这种方法的误差率较低(实验表明误差保持在 0.4W～2.4W 之间)，并且运行时的开销也较小. Joulemeter 机制的优点是不需要对应用程序负载或虚拟机里的操作系统进行额外的插桩，能自动适应应用程序的特征甚至硬件配置的变化. 而 Stoess 等人[21]提出的虚拟机能耗测量机制，有一个前提假设：即每个硬件部件的详细能耗模型是已知的，但实际上这种模型很难提供. McIntosh-Smith 等人[24]则针对异构系统提出了一种性能和能耗的基准测试方法.

2.2.2 能耗模型精度的挑战

在虚拟化云计算数据中心中，通常是通过评估当前虚拟机的 CPU 利用率来确定虚拟机的能耗情况. 但是不幸的是，这些评估可能并不精确[25]. 因为测量出来的 CPU 利用率，无法精确反映真实的 CPU 使用情况，它还包括了内存等待时间，因此不能真实反映 CPU 功耗情况. 例如，有两个虚拟机，一个是 CPU 密集，一个是内存密集，尽管两个虚拟机的 CPU 利用率测量值相同，如都是 100%，但是这两个虚拟机的能耗来源是不同的. 这证明了简单的虚拟机能耗模型存在局限性.

为了提高虚拟机能耗模型的精度，Krishnan 等人[25]研究了复杂的内存层次对能耗模型精度的影响，因为内存正在逐渐变为一个重要的部件. 能耗评估的精度依赖于虚拟机对内存系统的使用情况，如不同 Cache 层次的使用情况或者内存级的并行执行情况.

在云计算数据中心中精确捕获每个虚拟机的能耗不是一件容易的事. 这是因为今天云计算数据中心部署着各种类型的负载，它们有着不同的资源使用需求，并且经常动态变化. 此外，负责管理数据中心或云基础设施的操作者无法知道应用程序所在的虚拟机里的信息以及应用程序本身的执行行为. 没有人知道虚拟机上的应用程序正在做什么. 因此，需要采用一种黑盒技术来捕捉虚拟机的能耗情况，即轮廓分析的方法.

2.3 虚拟机能耗测量的步骤

虚拟机能耗测量的基本思路如下：(1) 首先建立一个能耗模型. 把特定类型资源的利用率(如 CPU)和整体系统能耗建立联系，为简单起见不考虑处在相对较低利用率层次的其它类型资源的能耗；(2) 使用轻量级监控工具测量每个虚拟机运行时的不同资源的利用率，例如可以通过典型虚拟化平台(如 Xen)提供的硬件性能计数器进行在线的轮廓分析；(3) 评估虚拟机能耗情况. 输入资源利用率，通过资源能耗模型的计算，间接推断虚拟机的能耗.

基于以上思路，系统的总能耗可以简单地用式(1)表示：

$$E_{\text{server}} = E_{\text{idle}} + E_{\text{cpu}} + E_{\text{mem}} + E_{\text{disk}} + E_{\text{net}} \quad (1)$$

每个系统子部件的能耗计算方法如下：CPU 子系统的精确能耗包括处理器能耗和 Cache 能耗. 可以通过监控每秒钟的指令执行次数来表示虚拟机 CPU 的使用率. 基于此就可以建立每秒钟指令数和动态服务器能耗的相互关系. 内存子系统的能耗模型也类似，可以通过每秒 Cache 的缺失率创建内存能耗模型，建立每秒钟的 Cache 缺失数和服务器能耗的关联模型.

2.4 虚拟机能耗轮廓分析方法

目前，大多数在线能耗监控都是基于物理节点的，如 Liu 等人[26]提出的 GreenCloud 架构，通过使用一个外部的功耗测量器来获得物理主机的总体能耗. 它只提供整个系统的能量消耗，而没有提供每个虚拟机的能耗，是一种粗粒度的能耗测量方法，能耗精度低. 为了提升能耗测量精度，需要开发细粒度的资源和能耗轮廓分析方法，不仅分析物理节点的能耗，还要分析每个虚拟机的能耗.

Bohra 等人[27]提出了一个新的能耗轮廓分析工具 VMeter，它对每个虚拟机的资源和能耗进行轮廓分析. 该工具对 CPU、Cache、Disk 和 DRAM 等系统子部件进行分别监控. Srikantaiah 等人[28]采用类似的方法，通过监控底层性能计数器预测上层应用程序或进程的能耗. 为了在软件设计时就考虑能耗的优化，需要对应用程序层的能耗轮廓进行分析. 为了达到这个目标，来自微软的研究人员提出了一个

自动化的工具进行细粒度的应用程序能耗轮廓分析，并提供有价值的信息给程序设计者[29]. Choi 等人[30-31]通过对服务器整合场景的能耗进行轮廓分析，开发出一个模型来预测整合场景下应用程序的平均和持久的能耗.

3　虚拟化云平台的能耗分析与建模

能耗模型可以分成两大类：基于系统功能单元的能耗评估模型和基于硬件性能计数器的能耗预测模型. 出于对模型精度的考虑，大多数的能耗模型属于第 2 类，即对通过硬件性能计数器收集到的关键系统事件进行统计，预测系统能耗，然后进行各种策略的能耗优化.

图 1 给出了针对虚拟化云计算平台的能耗建模方法，我们把它分为静态能耗建模和动态能耗建模两类. 静态能耗建模刻画的是单个虚拟机系统功能部件的能耗，可以分别对 CPU、内存、磁盘等系统子部件进行建模，也可以对虚拟机系统整体进行建模. 动态能耗建模，指的是对虚拟化动态应用场景的能耗进行建模，包括服务器整合能耗建模和在线迁移能耗建模.

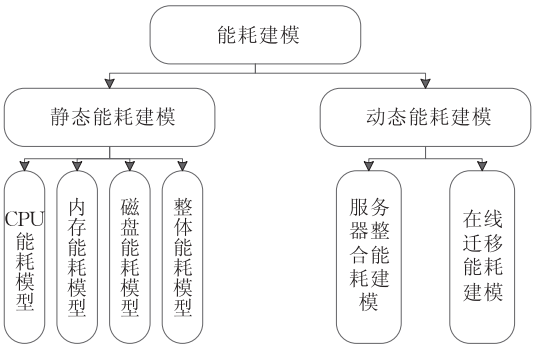

图 1　虚拟化云计算平台的能耗建模方法分类

3.1　虚拟机能耗模型

3.1.1　能耗与功耗的定义及相互关系

能耗和功耗是用来衡量系统能量消耗的两个重要概念. 在计算机系统中，能耗是指计算机系统一段时间内总的能量消耗，单位是焦耳(J). 而功耗是指单位时间内能量的消耗，反映计算机系统消耗能量的速率，单位是瓦特(W). 这两者都能表示系统的能耗情况，本文不作特别区别. 它们之间的关系如式(2)所示[32]

$$E=\int_{t}^{t+\Delta t} P \cdot \mathrm{d}t \tag{2}$$

3.1.2　CPU 能耗模型

CPU 的能耗使用模型依赖于多个因素，如处理器子单元的活动情况，执行特定指令的情况，片上 Cache 使用情况以及处于高低频的情况. 一个精确的能耗模型，需要考虑所有这些因素，但是这样会使得监控的开销很大，并且不适于运行时的虚拟机的实时能耗评估. 因此，Kansal 等人[23]提出了一个轻量级的替换方法来跟踪处理器的活动和休眠次数，这可以很容易地从操作系统里的处理器使用情况获得. 令 u_{cpu} 表示处理器利用率. 对于一个给定的处理器频率，CPU 的能耗模型如下[23]：

$$E_{\mathrm{cpu}}=\alpha_{\mathrm{cpu}} u_{\mathrm{cpu}}+\gamma_{\mathrm{cpu}} \tag{3}$$

其中，α_{cpu} 和 γ_{cpu} 指的是模型的特定常数，可以通过训练获得.

如果一个虚拟机 A 的处理器利用率表示为 $u_{\mathrm{cpu},A}$，那么该虚拟机的能耗 $E_{\mathrm{cpu},A}$ 为

$$E_{\mathrm{cpu},A}=\alpha_{\mathrm{cpu}} u_{\mathrm{cpu},A} \tag{4}$$

3.1.3　内存能耗模型

现有的内存能耗模型研究发现影响内存能耗的主要因素是内存读写的吞吐量. 尽管有用额外的插桩技术来捕获内存吞吐量[33]，也有一种轻量级的内存吞吐量的评估方法，即记录最后一层 Cache(Last Level Cache，LLC)的缺失次数，这在大多数处理器上很容易获得. 使用这些指标，内存的能耗模型可以被写为如下形式[23]：

$$E_{\mathrm{mem}}(T)=\alpha_{\mathrm{mem}} N_{\mathrm{LLC}}(T)+\gamma_{\mathrm{mem}} \tag{5}$$

其中，$E_{\mathrm{mem}}(T)$ 表示 T 时间内内存的总能耗，$N_{\mathrm{LLC}}(T)$ 表示 T 时间内 LLC 缺失次数，α_{mem} 和 γ_{mem} 表示线性模型的参数.

与跟踪虚拟机处理器使用率相比，跟踪虚拟机的 LLC 缺失没有那么直观. 因为内存访问由处理器硬件直接管理，操作系统和 Hypervisor 不能直接看到. 大多数的处理器把 LLC 缺失作为硬件的一个性能计数器，如 Intel Nehalem 处理器在每个核上都提供这个功能. 通过跟踪每个虚拟机在每个核上因上下文切换而导致的 LLC 缺失次数，我们就能获得相应虚拟机的 LLC 的缺失次数. 因此虚拟机的内存能耗模型如下[23]：

$$E_{\mathrm{mem},A}(T)=\alpha_{\mathrm{mem}} N_{\mathrm{LLC},A} \tag{6}$$

其中，$E_{\mathrm{LLC},A}$ 表示一个虚拟机 A 在时间 T 内在所有核上的 LLC 缺失次数，α_{mem} 含义与式(5)中一致.

3.1.4　磁盘能耗模型

磁盘子系统的能耗模型相对较难建立，这是因为无法知道磁盘的功耗状态，以及磁盘硬件缓存的

影响.在数据中心服务器中,磁盘大多数以 RAID(Redundant Array of Independent Disks,磁盘阵列)的方式存在,由 RAID 控制器控制着物理磁盘,Hypervisor 只能看到逻辑驱动.因此只能利用 Hypervisor 能看得到的参数来进行建模.而 Hypervisor 只能看到读和写的字节数,以及这些读/写的服务时间.因此可以使用这些参数来建立磁盘能耗模型[23]:

$$E_{\text{disk}}(T)=\alpha_{\text{rb}}b_{\text{r}}+\alpha_{\text{wb}}b_{\text{w}}+\gamma_{\text{disk}} \tag{7}$$

其中,$E_{\text{disk}}(T)$表示 T 时间内磁盘的能量消耗,b_{r}和 b_{w}表示 T 时间内读和写的字节数.α 参数和 γ_{disk}可以通过训练获得.

与 CPU 和内存资源一样,需要跟踪每个虚拟机的磁盘使用参数.值得注意的是,磁盘活动的时候,虚拟机不一定总是活动的,因为 Hypervisor 可能在进行批处理 I/O 中断,或者缓存 I/O 等操作.因此,需要在 Hypervisor 中显式的跟踪 I/O 操作,而不是观察虚拟机活动时候存储系统的活动情况.Windows Hyper-V Hypervisor 已经实现了大部分的状态跟踪操作,每个虚拟机特定的磁盘使用率能从 Hyper-V 的性能计数器里获得,它们是 Hyper-V Virtual Storage Device 和 Hyper-V Virtual IDE Controller.因此,可以得到以下的虚拟机磁盘能耗模型[23]:

$$E_{\text{disk,A}}=\alpha_{\text{rb}}\times b_{\text{r,A}}+\alpha_{\text{wb}}\times b_{\text{w,A}} \tag{8}$$

其中,$b_{\text{r,A}}$和 $b_{\text{w,A}}$表示虚拟机 A 读和写的字节数.此外,根据实验发现,可以忽略磁盘读和写的能耗差别,因此得到一个共同的参数,由前面提到的虚拟机磁盘计数器总和表示,表示读和写的总字节数,因此可以把模型简化为

$$E_{\text{disk}}(T)=\alpha_{\text{io}}\times b_{\text{io}}+\gamma_{\text{disk}} \tag{9}$$

相应的虚拟机的磁盘能耗为

$$E_{\text{disk,A}}=\alpha_{\text{io}}\times b_{\text{io,A}} \tag{10}$$

3.1.5 系统整体能耗模型

与式(1)表示的系统基本能耗模型不同,Bohra 等人[27]对监控事件的相互关联关系进行了研究,他们采用主成分分析(Principal Component Analysis,PCA)方法对输入数据集进行分析,发现{CPU,Cache}对和{Disk,DRAM}对有很高的相关性.因此,他们把系统负载分成两大类:CPU 密集的负载和 I/O 密集的负载.基于此,可以预测系统总功耗,其模型表达如下:

$$P_{\{\text{cpu,cache}\}}=a_1+a_2\,p_{\text{cpu}}+a_3\,p_{\text{cache}} \tag{11}$$

$$P_{\{\text{DRAM,disk}\}}=a_4+a_5\,p_{\text{DRAM}}+a_6\,p_{\text{disk}} \tag{12}$$

$$P_{\{\text{total}\}}=\alpha P_{\{\text{cpu,cache}\}}+\beta P_{\{\text{DRAM,disk}\}} \tag{13}$$

其中,a_1和 a_4是用来调节系统空闲时的系统能耗,参数 a_2、a_3、a_5和 a_6表示权重,P_{cpu}、P_{cache}、P_{DRAM}和 P_{disk}分别表示 CPU、Cache、DRAM 和 Disk 各子部件的功耗,可由监控到的系统事件计算得到.$P_{\{\text{cpu,cache}\}}$和 $P_{\{\text{DRAM,disk}\}}$分别表示{CPU,Cache}和{DRAM,Disk}子系统的功耗.使用这个模型,可以计算每个活动虚拟机的功耗.

整体系统功耗可以分为两大类:基数功率消耗和动态功率消耗.根据上式,a_1和 a_4提供系统空闲时的基数功率消耗,因此基数功耗表示如下:

$$P_{\{\text{baseline}\}}=\alpha\times a_1+\beta\times a_4 \tag{14}$$

其中,$P_{\{\text{baseline}\}}$是基数功率消耗.系统整体功率消耗表示如下:

$$P_{\{\text{total}\}}=P_{\{\text{baseline}\}}+\sum_{k=1}^{N}P_{\{\text{domain}(k)\}} \tag{15}$$

其中,$P_{\{\text{total}\}}$表示总体系统功率消耗,$P_{\{\text{domain}(k)\}}$表示一个活动虚拟机域的功率消耗,N 是活动虚拟机域的个数.每个虚拟机域可以由系统提供的硬件事件的计数器值获得

$$P_{\{\text{domain}(i)\}}=\alpha(a_2\,p_{\text{cpu}(i)}+a_3\,p_{\text{cache}(i)})+\beta(a_5\,p_{\text{DRAM}(i)}+a_6\,p_{\text{disk}(i)}) \tag{16}$$

其中,$P_{\{\text{domain}(i)\}}$是指一个活动虚拟机或 Dom0 的功率消耗,$p_{\text{cpu}(i)}$、$p_{\text{cache}(i)}$、$p_{\text{DRAM}(i)}$可以从给定域的硬件事件计数器值获得,$p_{\text{disk}(i)}$可以从给定域的硬盘数据传输量获得.实验证明,该模型能很好地满足相应的测试基准程序的测试,计算开销也不大,并且提升了评价预测的精度.

3.2 服务器整合建模

服务器整合指的是把多个虚拟机应用负载整合在一台物理主机上运行[10,12,34],它是虚拟化技术的一个重要应用场景.服务器整合可以被形式化为如下形式[35]:令在 T 时间段,有 N 个程序 A_i需要被部署到 M 个物理主机 H_j上.对于每个程序 A_i,令 $C(A_i,t)$表示在 t 时刻为了满足 SLA 所需的资源.这里假设 $C(A_i,t)$可以从资源的监控数据获得,而不去考虑怎样把程序的 SLA 转为一个资源值.令物理主机 H_j的容量为 $C(H_j)$,X 表示一个特定的整合配置,把程序部署到物理服务器上,如果程序 A_i被部署到主机 H_j,那么令 $x_{ij}=1$,否则 $x_{ij}=0$.服务器整合问题就是要找这样一个配置,优化给定的开销函数.例如,如果整合的目标是优化能耗,那么需要寻找一个配置 X,最小化 $P(X)$,$P(X)$是一个真实估价的函数,表示一个特定程序部署的能量消耗.部署应该确保所有程序的资源需求满足整个 T 时间

段，即 $\forall t\in T,\sum_{i=1}^{N}x_{ij}C(A_i,t)\leq C(H_j)$. 我们还需要确保所有的应用程序被成功部署上，即 $\sum_{j=1}^{M}x_{ij}=1$.

动态整合假设 T 非常短，这使得应用程序的容量请求 $C(A_i)$ 是时间无关的. 因此，容量约束不再是随机的. 在动态整合场景里，为了评估 $C(A_i)$ 和 $C(H_j)$，一个主流的指标是来自 IDEAS(Ideas International)组织的 RPE2(Relative server Performance Estimate 2)指标①，几乎所有的常用服务器的性能都用 RPE2 值测试出来. $C(H_j)$ 用 RPE2 值来表示，通过分析服务器 CPU 的利用率对应用程序的资源请求情况进行评估. 如果没有使用虚拟化，服务器的 RPE2 值和该时间段内服务器的最大 CPU 利用率的乘积，被用来表示应用程序的资源需求大小. 如果使用了虚拟化，则根据其物理服务器上的每个虚拟机的权重、虚拟机的 CPU 利用率和物理服务器的 RPE2 值进行综合计算.

3.3　在线迁移建模

虚拟机在线迁移指的是一种在不停机的情况下进行虚拟机迁移的技术[11,36-37]. Liu 等人[38]对虚拟机迁移的能耗进行量化建模. 虚拟机迁移的能量消耗主要由数据传输率决定. 源端主机在进行虚拟机迁移时的能量消耗随数据传输率的上升而上升. 另一方面，当数据传输率更高的时候，迁移延迟变得更短. 实验证明，因为虚拟机迁移本身而产生的能量消耗与数据传输率无关. 注意到，虚拟机迁移是一个 I/O 密集的程序，能耗主要消耗在网络上进行数据传输和接收. 可以推测虚拟机迁移导致的能量开销只由网络流量的数据量决定. 基于此，Liu 等人设计了一个模型来评估虚拟机迁移的能量开销.

实现虚拟机迁移牵涉到源主机、网络交换器、目标主机 3 大资源. 因为交换结构非常复杂，能耗很难量化，因此该模型只考虑源端和目标端的能量消耗. 一般来说，数据在源端传输的量与数据在目标端接收的量是相等的，另外通过实验发现在同构的环境中，数据传输和接收的能量消耗差异非常小. 因此以下能耗模型假设能量消耗与虚拟机迁移导致的网络开销成线性增长关系：

$$E_{\mathrm{mig}}=E_{\mathrm{source}}+E_{\mathrm{dest}}=(\alpha_s+\alpha_d)V_{\mathrm{mig}}+(\beta_s+\beta_d)\quad(17)$$

其中，α_s、α_d、β_s、β_d 表示需要训练的模型参数. 在异构物理机环境下，该模型仍然适用，只是这些模型参数需要重新训练. 因为当前的虚拟化平台(包括 Xen 和 VMware)只支持虚拟机在同构主机间的迁移，为了简化问题，只在同构环境中训练能耗模型. 因此，式(17)能被简化为

$$E_{\mathrm{mig}}=E_{\mathrm{source}}+E_{\mathrm{dest}}=\alpha V_{\mathrm{mig}}+\beta\quad(18)$$

网络流量 V_{mig} 用兆字节(Mb)表示，能耗 E_{mig} 用 Joules② 表示.

V_{mig} 可以很容易从真实迁移训练实验获得. 为了计算额外的能耗 E_{mig}，应该获得由虚拟机在线迁移导致的能耗上升部分. 首先测量物理主机的静态功耗 P_0，然后测量当虚拟机在迁移时的能耗 P_t，那么由虚拟机迁移导致的动态能耗可以由下式计算得到

$$E_{\mathrm{mig}}=\int_0^{T_{\mathrm{mig}}}(P_t-P_0)\,\mathrm{d}t=\sum_{t=1}^{\lceil T_{\mathrm{mig}}\rceil}(P_t-P_0)\quad(19)$$

4　虚拟化云平台能耗管理实现机制

节能技术可以分为静态节能技术和动态节能技术. 第 1 种方法在最初系统和部件设计的时候就把能耗因素考虑进去，这种方法包括电路层节能技术(如对未使用的组件禁用时钟信号)，节能微架构设计[8]，处理器、内存和磁盘的低功耗状态设计[39]等. 第 2 种方法根据负载变化，从资源管理的角度自适应地进行节能管理，这种方法需要智能的管理软件来优化服务器层和集群层的能量消耗. 把软硬件节能技术绑定在一起是一种更为有效的方法，也是虚拟化云计算平台可行的节能实现机制.

根据之前的分析，虚拟化云平台的能耗管理面临一些新的挑战，因此需要针对该平台开发新的能耗管理机制. Nathuji 和 Schwan[19]考虑了面向虚拟化平台的能耗管理新机制的设计问题：(1) 软硬结合的能耗调整. 虚拟化技术可以动态扩展物理资源分配给虚拟机，有两种方式可以实现能耗调整：① 软的技术，依靠 Hypervisor 的管理来限制虚拟机对硬件资源的使用率；② 硬的技术，采用底层硬件节能技术，如处理器电压频率调整(Dynamic Voltage and Frequency Scaling，DVFS)技术. 为了实现更有效的虚拟化层能耗管理，这两方面的技术都要采用. (2) 独立和协调. 客户虚拟机有自己的能耗管理方法，如 Linux 操作系统允许处理器进行动态电压频率调整，这个管理策略既可以装载到内核，也可以在用户空间执行. 此外，还可以采用其它特定应用

① http://www.ideasinternational.com/performance/
② http://en.wikipedia.org/wiki/Joule

程序的节能策略解决实时负载或者满足程序最小化能耗的需求.从这个角度来说,云平台能耗管理的一个必要元素是需要协调各个层次的节能策略,如硬件层、操作系统层、机架层、数据中心层等.(3)管理的灵活性.有虚拟化技术支持的现代数据中心拥有多种设备,这些设备有不同的属性和管理功能,或者部署了不同类型的应用程序,而且这些应用程序有着不同的 SLA 需求.这种场景下需要不同的动态能耗管理策略.因此为了有效解决虚拟化环境的能耗管理问题,需要给管理员提供灵活的能耗管理策略.

根据管理层次的不同,我们把虚拟化云计算平台的能耗管理机制分为两大类:虚拟化层的能耗管理机制和云平台层的能耗管理机制.其中在虚拟化层的能耗管理机制中,我们分析了虚拟机管理器(Virtual Machine Monitor,VMM)层对能耗管理的支持,以及虚拟机层对能耗管理的支持.云平台层的能耗管理机制范围很广,它的主要思路是从虚拟资源管理的角度对云平台的能耗进行管理,我们又把它具体分为 VPM(VirtualPower Management)能耗管理机制和能耗管理中间件框架两大类,如图 2 所示.

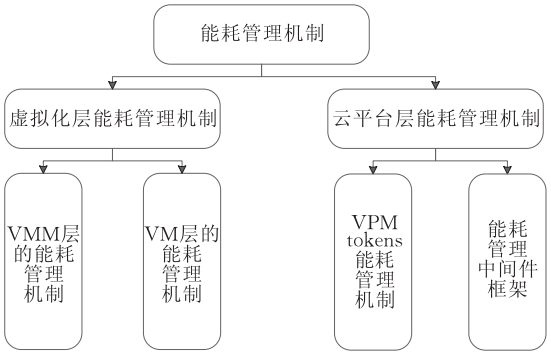

图 2 虚拟化云计算平台的能耗管理机制分类

附录表 1 列举了目前各种典型的虚拟化云计算平台的能耗管理机制,分别从实现平台、考虑的资源维度、同构/异构性、是否用到迁移、最终优化目标、采用的具体节能技术和实现效果 7 大方面,全面总结、分析和比较了各种能耗管理机制的异同点.

4.1 虚拟化层的能耗管理机制

4.1.1 VMM 层的能耗管理机制

虚拟机管理器(VMM)可以有两种方式参与能耗管理[55]:(1)虚拟机管理器可以看作为一个能耗感知的操作系统,对系统整体性能进行监控,并利用 DVFS 等技术降低系统部件的能耗;(2)依靠操作系统特定的能耗管理策略和应用程序级的信息,把不同虚拟机的能耗管理操作映射到硬件功耗状态的真实改变上.虚拟机管理器一般提供类似于 Linux 本身提供的按需能耗管理机制,即支持基于高级配置与电源接口(Advanced Configuration and Power Management Interface,ACPI)的能耗管理机制.系统间隔性的监控 CPU 利用率,检测适合的功率状态,生成一个平台独立的命令,进而调节硬件的功率状态.

Xen 支持 ACPI 的 P 状态,并在 cpufreq 驱动里实现这一机制[56].与 Linux 功耗管理子系统类似,Xen 功耗管理系统包括 4 个管理器:(1)按需管理器.根据当前的资源需求选择最佳的 P 状态;(2)用户空间管理器.由用户指定设置 CPU 的频率;(3)性能管理器.设置最高的可用时钟频率;(4)节能管理器.设置最低的时钟频率.除 P 状态外,Xen 还支持 C 状态(CPU 睡眠状态)[56].当一个物理 CPU 没有运行任务时,就切换到 C 状态.当新的请求到来时,CPU 切换回活动状态.一个问题是进入到哪一种 C 状态:深度 C 状态提供更高的节能效果,但是也意味着更高的切换开销.目前,Xen 缺省地把 CPU 切换到第一个 C 状态.当 CPU 收到唤醒信号时,会不可避免地带来一定的性能损失.与 Xen 相似,VMware 也支持主机层的能耗管理机制,支持 DVFS 技术.系统连续监控 CPU 使用率,并恰当地使用 ACPI 中的 P 状态.KVM 是另外一种虚拟化平台,它是作为 Linux 内核的一个模块.在这个模块下,Linux 担任 Hypervisor 的角色,所有的虚拟机按正常的进程由 Linux 调度器进行调度.这种方法减少了 Hypervisor 实现的复杂度,因为调度和内存管理都是由 Linux 内核完成.KVM 支持 S4(休眠)和 S3(待机)两种功耗状态①.S4 不需要 KVM 任何特殊的支持,在休眠状态,客户操作系统把内存状态保存进硬盘,并且关闭计算机.在下次启动的时候,操作系统从磁盘读取保存着的内存状态,从休眠中恢复,初始化所有设备.在 S3 状态,内存是开着的,数据不需要保存到磁盘.但是,客户操作系统必须保存设备的状态,因为他们需要恢复设备状态.在下一次启动时,BIOS(Basic Input Output System)应该识别 S3 状态,而不是初始化设备,直接恢复 S3 保存着的设备状态.因此 BIOS 需要做一些修改来支持这种行为.

在虚拟机管理器层,除了对 ACPI 的支持之外,另一个重要的特征是对虚拟机在线迁移的支持.通

① http://www.linux-kvm.org/page/PowerManagement

过这种虚拟机的迁移可以进行能耗感知的动态虚拟机整合，进而达到节能的目的. 迁移被用来在物理主机间转移虚拟机. 离线迁移采用暂停(Suspend)的方式把一个虚拟机从一台主机移动到另外一台主机，拷贝内存内容，然后在目标主机恢复运行虚拟机. 而在线迁移转移虚拟机时，不需要暂停虚拟机. Xen 既支持离线迁移也支持在线虚拟机迁移. VMware 的 VMotion 模块使得虚拟机能通过自动的或者管理员手动的方式在物理节点间在线迁移虚拟机. VMware 的 DRS (Distributed Resource Scheduler)模块包含一个专门的功耗管理子系统叫 VMware DPM(Distributed Power Management)①，用来动态关闭空闲服务器，从而减少能量的消耗. 当资源请求量上升的时候，服务器又被重新启动起来. VMware DPM 使用在线迁移来重新分配虚拟机，保持最小的活动服务器数目. KVM 也支持在线迁移.

表 2 对目前流行的 3 大虚拟机管理器(Xen、VMware 和 KVM)的能耗管理机制进行了比较分析.

表 2　典型虚拟机管理器的能耗管理机制比较

VMM 类型	VMM 能耗管理机制	基于迁移的能耗管理机制
Xen	支持 P 状态、C 状态	基于在线迁移的整合
VMware	支持 P 状态	基于 VMotion 迁移的整合、DPM 能耗管理机制
KVM	支持 S4、S3 状态	基于在线迁移的整合

4.1.2　VM 层的能耗管理机制

在虚拟机里进行能耗管理面临很多挑战，Nathuji 等人[19]提出的虚拟能耗管理机制 VirtualPower 填补了在虚拟机内部进行能耗管理的空白，它结合采用硬件能耗调整和基于软件的能耗调整方法来控制虚拟化平台的能耗. VirtualPower 定义了一整套完整的虚拟能耗管理(VirtualPower Management，VPM)元素：VPM 状态、VPM 通道、VPM 规则、VPM 机制. 通过虚拟机层的“软的 VPM 状态”管理策略，客户机会拥有一致的硬件管理能力视野，而不用考虑底层的物理资源. 然后通过 VPM 规则把这些更改映射到底层硬件的改变. VirtualPower 把能耗管理策略解释成一种提示(hints)而不是执行命令. 这些提示被 ACPI 接口捕获. 当客户虚拟机想要通过这个接口进行特权操作时，目前的 Hypervisor 的做法是会忽视它们，但是 VirtualPower 会解释提示并把它们映射到 VPM 通道. 这些通道提供有用的提示给 VirtualPower 的 VPM 规则. 然后，VPM 规则使用由这些提示组成的“软”的状态请求作为局部能耗管理和全局能耗管理的输入. VPM 状态和通道的绑定提供了一致的能耗管理，同时维护了虚拟机的隔离性和独立性.

VirtualPower 不需要修改客户操作系统. 它的架构流程如图 3 所示：每个物理平台运行一个 Hypervisor 及配套的 Dom0. 客户虚拟机基于“软”的 VPM 状态进行能耗管理. VPM 通道捕获客户虚拟机的能耗管理请求，通过在 Hypervisor 更新这些“软”的状态. 这些信息随后被传递到能耗管理软件部件，这些部件由运行在 Dom0 的 VPM 规则集合组成. 最后采用 VPM 机制来真实执行能耗管理决策.

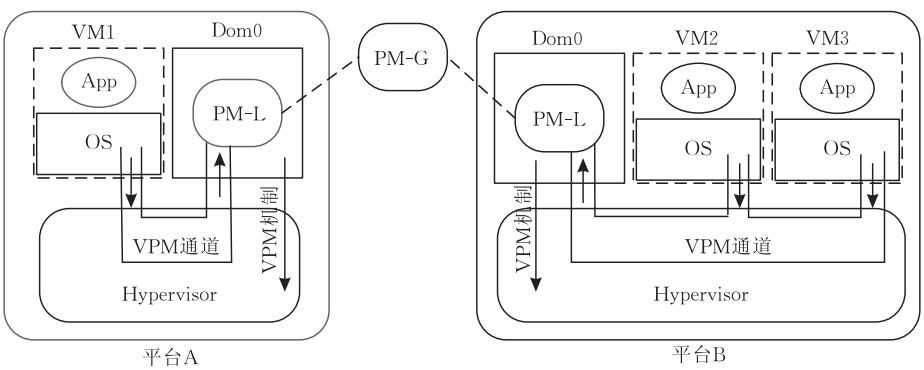

图 3　VirtualPowever 能耗管理机制架构[19]

4.2　云平台层的能耗管理机制

4.2.1　VPM tokens 管理机制

前面提到的 VPM 机制实现了在虚拟机里进行能耗管理操作的基本功能，这是进一步实现复杂在线能耗管理的基础. 这种机制为底层平台能耗管理方法的多样性提供了一个统一的管理方式. VPM 支持 3 种复杂能耗管理机制：硬伸缩、软伸缩和整合[19].

① http://www.vmware.com/pdf/vsphere4/r40/vsp_40_resource_mgmt.pdf

(1)硬伸缩(Hardware Scaling).不同的平台和设备架构有不同的硬件伸缩能力.这些伸缩功能依赖于虚拟机层的资源共享情况(如虚拟机运行在多个核上).VPM 机制支持硬伸缩,允许 VPM 规则在客户操作系统运行的时候进行硬件状态的设置,通过 VirtualPower 超级调用接口设置 VPM_SET_PSTATE.当然,规则也会判断这些状态改变是否容易实现.例如,策略规则必须判断是否会因为资源共享的关系导致虚拟机的硬件性能状态产生冲突.在处理器动态电压频率调整的时候,这些冲突通过硬件解决,确保给芯片提供足够的电压来运行最高频率.

(2)软伸缩(Soft Scaling).硬伸缩不总是有效,有时只提供很小的收益.因此,提出了"软"资源伸缩的概念,通过资源调度的方式来模拟硬伸缩动作带来的性能损耗.对于处理器的管理,可以通过修改 Hypervisor 的虚拟机调度属性来模拟虚拟机希望获得的性能模式.例如,如果一个虚拟机根据之前的性能状态请求把一个核缩小到半个核,这个时候执行软伸缩,Hypervisor 调度器可以把客户机的最大时间片降低到原来的一半.调度参数的调整是通过在 VirtualPower 超级调度接口上进行 VPM_SET_SOFT 设置来实现.软伸缩能带来很好的节能效果.当处理器处于空闲状态时,软伸缩很有效.同时,通过在多个资源上正确地管理软伸缩,如协调并发或者使用整合,也能获得额外的节能效果.

(3)整合(Consolidation).在多个资源上(如多核芯片上的核)进行多个虚拟机的软伸缩,会导致共享资源的处理器核负载不平衡,可能有些核空闲,有些核满载.资源整合的第一个优点是大幅节能,通过把没有负载运行的资源切换到空闲或暂停状态.第二个优点是可以考虑资源异构性.特别的在数据中心环境,可能存在针对特定负载更高能效的服务器.通过绑定软伸缩与虚拟机重映射或者迁移,可以把多个兼容的虚拟机实例映射到适合的高能效物理资源上.

Nathuji 等人[20]进一步提出了一套集群层和数据中心层的管理部件,他们提出的虚拟机感知的能耗预算把多个分布式管理器整合到 VirtualPower 管理框架中[19],目标是在一定的能耗预算下最大化性能或效用.通过实现 4 个系统功能来达到这一目标:(1)以虚拟机为中心的预算(VM-Centric Budgeting).提出了一个按比例分配的模型,按虚拟机的相对效用(重要性)来分配能耗预算.(2)应用感知的管理(application-aware management).在维持能耗预算的同时,提供足够的性能.(3)预算补偿应用(compensating budgeted applications).当一个虚拟机处于低功耗的时候,可以把功耗省给别的虚拟机用,当虚拟机需要而别的又处于低功耗的时候,可以获得补偿.(4)预算异构资源(budgeting heterogeneous resources).在异构平台中,当同时降低相同能耗数时,如 10 W,各平台的性能下降幅度是不一样的.基于此,可以选择最合适的平台进行降低功耗操作.

4.2.2 能耗管理中间件框架

针对模块化和多层操作系统结构,Stoess 等人[21]提出了一种创新的能耗管理框架.这种框架提供了一个统一的模型进行能量分割和部署以及进行能量感知的资源记账和分配.他们基于 Hypervisor 的虚拟机系统实现了一个系统原型,该原型由两部分组成:(1)主机层子系统.控制整个机器的能量约束,并最大化所有的客户操作系统和服务部件的能效.(2)能量感知的操作系统.针对特定程序进行细粒度的能量管理.实验证明,对于能耗感知的操作系统和普通的操作系统,这个框架都能精确控制和保证每个硬件设备的能耗.

Jung 等人[40]提出了一个整体控制框架 Mistral,优化能耗,提高性能收益,减少由于各种操作和控制器本身带来的短暂的开销,来最大化整体效用.Mistral 通过多个可扩展的优化算法,解决了不同分布式应用程序和大规模基础设施的能耗管理问题.最近 Oh 等人[57]研究了虚拟机共存和 CPU 热量管理的影响,探索性能干扰,并且基于实验结果,设计了一个负载感知的虚拟机在线调度器进行云计算环境的分布式能耗管理.其它还有一些虚拟机管理框架,解决了虚拟机在多物理机上的部署问题,最小化物理机个数,达到适应当前负载的目的[10,58].

云计算数据中心的一个关键目标是最大化收益、最小化能量消耗和主机程序的 SLA 冲突.Van 等人[59]提出了一个资源管理框架,由一个基于效用的动态虚拟机供给管理器和一个动态虚拟机部署管理器组成.这两个问题都被建模成约束满足问题(constraint satisfaction problem).虚拟机供给的目标是最大化全局效用,同时满足 SLA、最小化云计算基础设施能量相关的操作开销.另外,Liu 等人[26]提出了 GreenCloud 架构,目标是降低数据中心的能耗,同时保证性能.GreenCloud 架构允许在线监控、在线虚拟机迁移、虚拟机优化部署等操作.

4.3 性能与能耗的权衡分析

虚拟化云计算平台中一种常用的节能方法是通过虚拟机在线迁移技术进行能耗感知的服务器整合，从而腾出空闲的服务器，关闭或转为低功耗状态运行.但服务器整合同时会影响程序性能或服务质量[60-62]，尤其在分布式的在线服务中要小心使用，如在线购物、企业程序等，因此需要权衡性能与能耗之间的关系.因为负载是动态变化的，需要进行运行时的整合活动，而且迁移不是免费的，要依赖于负载类型.因此，除了考虑服务器整合内在的能耗和性能权衡，基础设施提供者还要考虑进行资源重配置的收益和开销的权衡[63].考虑到这种重配置引起的开销，需要重新思考最佳的节能策略.例如，当负载变化很快时，损失掉一些性能比触发一个昂贵的迁移更好，否则迁移的成本可能还没有被收回，下一个迁移又开始了.或者采用简单且适中的变化，如对虚拟机重新分配资源，比启动新的主机更加有效.最后，能耗开销和决策延迟导致的开销也需要考虑.

Chase 等人[64]使用一个经济模型来进行集群中功耗感知的资源分配.Kephart 等人[65]也实现了一个控制器架构，用户可以指定功耗和性能目标.Park 等人[66]提出了一个针对在多核处理器上运行多线程程序的有性能保障的能耗管理方法，节约能耗的同时减少了性能的损耗.Nathuji 等人[20]提出了一个框架来维持虚拟机的功耗和性能权衡，并且在功耗预算下有效地管理物理机，获得好的性能.Ye 等人[45]针对虚拟化云计算数据中心环境，从实验评估的角度探索了通过虚拟机迁移和服务器整合获得的能耗减少与迁移和整合本身的性能开销的权衡关系.

Gandhi 等人[67]从理论分析的角度出发，采用能量和响应时间乘积(ERP)这个广泛使用的指标来计算能量与性能的权衡结果，并且给出了服务器群(Server Farm)能耗管理策略的最优理论结果.对于一个固定的资源需求模式，他们证明存在一个非常小的自然策略集合，对于单个服务器，总是包含最优策略，并且推测对于多服务器系统，包含近似最优的策略.对于随时间变化的请求模式，他们给出了一个简单的流量无关的策略，为近似最优策略提供分析和实证证据.Beloglazov 等人针对云计算应用程序的多样性和动态可变性，提出虚拟机部署应该以在线的方式连续优化，为了了解问题的在线性质的影响，进行竞争分析，针对单虚拟机迁移和动态虚拟机整合问题，证明最佳的在线确定性算法的竞争比.此外，他们还基于虚拟机资源使用的历史数据的分析，提出了一种自适应的启发式算法进行虚拟机的动态整合，该算法能大幅减少能量的消耗，同时满足 SLA[68].

5 虚拟化云平台能耗管理算法

5.1 能耗管理算法的分类

从随机算法到基于学习的算法，能耗管理算法的范围很广[69].附录表 2 从不同角度对能耗管理进行总结、分类和比较，并给出了典型案例.

5.1.1 按主被动模式分

按照主被动模式分，能耗管理算法可以分为主动节能方法和被动节能方法.主动节能算法指的是通过对历史数据学习等方法，对未来的能耗情况进行预测，并预先根据预测信息进行功耗感知的资源管理.被动节能算法指的是通过实时监控等手段，根据当前的资源使用情况，进行相应的资源调整，达到节能的目的.

Bradley 等人[70]使用短期和长期负载预测提出了两个主动功耗管理算法.这两个算法根据临时负载模式，积极主动地提供足够可用的资源，达到节能的目的.Chen 等人[71]研究了负载到达流量模式，并使用它们来预测所需的资源总量，据此进行负载的能耗管理.Choi 等人[31]开发了一个模型来预测整合场景下应用程序的平均功耗和持久功耗，并在 Xen 平台上进行了实验评估，实验表明该方法预测平均功耗的误差在 5% 以内，持久功耗的误差在 10% 以内.

被动节能算法包括：Liu 等人[26]提出的基于监控的节能架构 GreenCloud、Bohra 等人[27]提出的基于系统部件监控的能耗轮廓分析工具 VMeter 以及 Ye 等人[45]提出的基于资源监控的虚拟数据中心节能架构等.

5.1.2 按算法精度分

按照算法精度分，能耗管理算法分为基于控制论的精确算法和基于启发式的算法.

采用控制理论进行能耗管理由来已久.Lefurgy 等人[84]证明了一个控制理论解决方案与一个常用的启发式解决方案相比，拥有更精确的功耗控制和更好的性能.Wu 等人[85]通过控制多时钟域处理器上的同步队列来管理功耗.采用控制理论进行虚拟资源管理的研究工作还有文献[86-88]，但他们只考虑了性能的控制，没有考虑功耗的控制.

Kusic 等人[72]把虚拟化异构环境中的能耗管理问题定义为一个连续优化问题,并采用有限的超前控制方法解决,目标是解决最大化资源提供者的收益、最小化功耗和 SLA 间的冲突. 最近,Wang 等人[73-74]针对虚拟化的服务器集群,提出了一个创新的基于反馈控制理论的集群层控制架构 Co-Con,协调来自不同硬件/软件制造商的功耗和性能控制策略. 实验结果表明 Co-Con 能同时对程序级的性能和底层的功耗提供有效的控制. 同时,他们还基于控制理论提出了一个两层控制架构 PARTIC,解决了虚拟化共享平台上的能耗控制问题[75]. Raghavendra 等人[76]采用控制理论和应用反馈控制环路来绑定和协调五个不同的功耗管理策略.

但是,基于控制理论提出的模型也存在一些缺点. 如当程序调整的时候,需要重新基于模拟的学习,这对于基础设施即服务(IaaS)的云提供者(如 Amazon EC2)来说是不现实的. 此外,模型也存在着固有的复杂度,主要体现在执行时间上. 例如,对于 15 个节点来说,优化控制器本身的执行时间就高达 30 min[72],这对于现实中大规模的云计算系统来说是不可接受的.

基于启发式的能耗管理算法并不需要在程序部署前进行基于模拟的学习,在真实的大规模云计算系统中可以获得较高的性能. 目前,虚拟化云平台的绝大多数自适应能耗管理算法都是基于启发式的[19-21,26,30,57-59].

5.1.3 按算法粒度分

按照算法的粒度分,能耗管理算法可分为细粒度算法、粗粒度算法以及粗细粒度混合算法.

细粒度的能耗管理算法指的是在单机或者系统部件层次的能耗管理. Rajamani 等人[77]提出了一个功耗感知的请求分配系统,用来解决数据中心的功耗问题. 他们把独立的功耗请求作为基本的功耗调度工作单位. 在这个环境中,关闭一个节点时不需要迁移或备份,这是一个细粒度的电源管理策略. 此外,也有一些研究人员针对多核处理器的能耗进行细粒度的管理,如从 CPU 部署和节点部署角度进行的细粒度的部署研究[78-79],允许作业被部署到可用处理器的子集上,而把剩余的关闭.

与前面的工作不同,粗粒度的能耗管理关注的基本工作单位更大一些. Lim 等人[53]把虚拟机域作为一个基本的分配单元,提出了一种叫做功耗感知的域部署(PADD)方法. 该方法依靠虚拟机动态迁移技术,把虚拟机域尽可能地部署到更少的物理机上,从而达到节能的目的. 此外,他们还开发了一个两层自适应缓存机制进行容量的预留,以此来避免与 SLA 的冲突.

也有一些研究人员把细粒度的方法和粗粒度的方法综合起来使用,提出一种粗细粒度混合的算法,如 Nathuji 等人[19]提出了一个局部和全局策略混合的数据中心资源管理系统的架构. 在局部层,系统利用客户操作系统的能耗管理策略. 全局管理器则获得来自局部管理器的当前资源分配的信息,并应用这个策略来决定是否需要采用新的虚拟机部署方案.

5.1.4 按资源种类分

按照资源的种类分,能耗管理算法分为计算资源节能方法、存储资源节能方法和网络资源节能方法.

有关计算资源节能的研究工作非常多,目前大多数的能耗管理算法都是针对计算资源的,如 DVFS 等机制以及通过虚拟机迁移达到节能目的的方法[19-21,41-42,45,47,51,53].

Ye 等人[54]针对存储虚拟机的磁盘能耗情况,提出了 3 种优化磁盘能效的机制:虚拟机管理器缓冲、早期冲刷和有缓冲支持的早期冲刷. 实验表明能有效优化虚拟机存储磁盘的能效. Kaushik 等人[49]针对 Hadoop 分布式文件系统提出了一种新的技术 GreenHDFS,把 Hadoop 集群分成逻辑上的热点区(Hot Zone)和冷点区(Cold Zone),通过 GreenHDFS 技术增加了服务器的冷点区数量,然后处于冷点区的服务器的 CPU、内存、磁盘可以被转为低功耗模式,从而达到有效节能的目的.

网络资源的节能是另一个重要的方向. Lin 等人[32]对网络的节能机制进行了综述,分析比较了不同的网络节能方法. Seetharaman[50]针对虚拟化环境,提出了虚拟化网络能耗在多租户网络中的节能方法,通过刺激每个租户来实现节能,租户可能重新调整它们的活动来最小化网络资源的能耗.

5.1.5 按应用场景分

虚拟化技术作为一种关键的底层支撑技术应用在云计算数据中心中,因此对云计算数据中心中的虚拟化应用场景进行能耗研究非常必要. 对这些创新场景的能耗管理,在传统的服务器能耗管理活动中是没有考虑到的. 下面,我们重点对这些特有的虚拟化云计算平台应用场景的能耗管理算法展开分析.

按照应用场景分,能耗管理算法分为 3 大类:节

能部署算法、节能整合算法和节能迁移算法，如图 4 所示. 节能部署算法可以分为静态部署算法和动态部署算法. 静态部署算法包括基于相关性的部署算法(CBP)和基于峰值聚类的部署算法(PCP). 动态部署算法包括能耗最小化部署算法(mPP)、历史感知的部署算法(iFFD)和迁移开销感知的部署算法(pMaP). 节能整合算法分为 ECTC 部署算法和 MaxUtil 部署算法. 节能迁移算法又细分为：单阈值迁移算法(ST)、最小化迁移算法(MM)、最高增长潜力算法(HPG)和随机选择算法(RC).

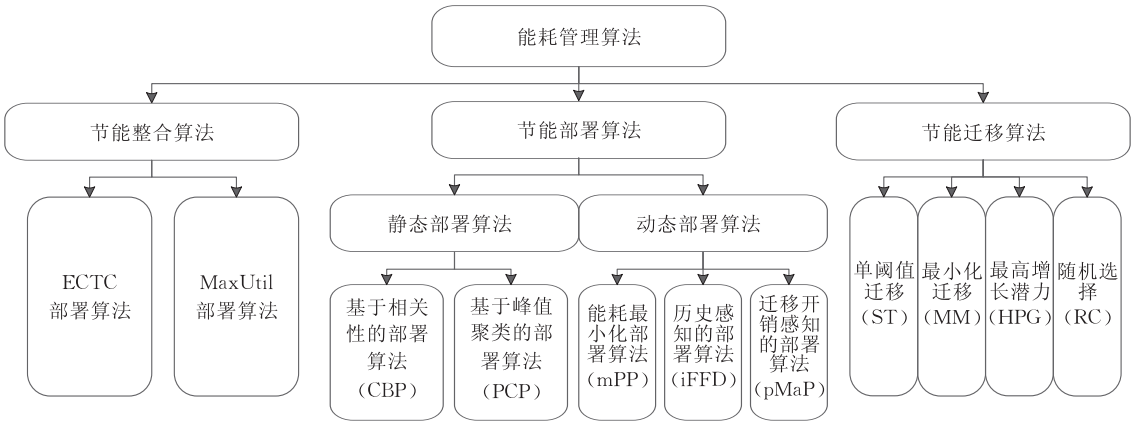

图 4　虚拟化云计算平台的能耗管理算法分类

5.2　节能部署算法

能耗感知的部署算法(Power-Aware Placement Algorithm)可以抽象为一个装箱问题，目标是使得能耗最小. 当数据中心配置发生改变(如添加了额外新的负载)或者负载模式发生变化(如某些负载的程序流量发生变化)时，会重新计算部署. 最小化功耗的装箱有如下特征：(1) 使用尽可能少的服务器来装这些虚拟机；(2) 优先选择能量使用效率高的服务器；(3) 选择跟自己容量相近的服务器，避免碎片；(4) 在重配置过程中最小化迁移次数.

虚拟化数据中心的负载部署包括静态部署和动态部署. 静态部署方法主要是出于方便性的考虑而不是效率. 早期的负载管理工具主要功能是监控和报告当前的状态给管理员，管理员再根据报告决定合适的负载管理策略. 容量规划就是一种典型的静态初始部署活动，比较著名的容量规划工具有 Cirba Data Center Intelligence、VMware Capacity Planner 和 Platespin PowerRecon 等. 也存在一些动态管理的工具，如 VMware Distributed Resource Scheduler(DRS). 目前有一些工具支持自动制定最优的数据中心负载配置，但这些工具大都基于蛮力搜索，而没有考虑程序敏感性及虚拟机共存等开销[89].

5.2.1　部署问题的形式化描述及其目标函数

虚拟机部署问题可以描述为把虚拟机分配矩阵 $\boldsymbol{R}_g=\{R_{11},\cdots,R_{ij},\cdots,R_{ms}\}$ 分解为虚拟机集合 $V=\{VM_1,\cdots,VM_l\}$ 和物理主机集合 $H=\{H_1,\cdots,H_n\}$. 部署问题的目标决定了部署矩阵 $\boldsymbol{P}$，如果虚拟机 v 被分配到主机 h 上，则 $P_{hv}=1$，反之 $P_{hv}=0$. 任何的部署解决方案必须满足物理主机的容量约束：

$$\forall h\in\{1,\cdots,n\}\sum_{v=1}^{l}P_{hv}\times\mathrm{CPU}(V_v)\leq\mathrm{CPU}(H_h)\quad(20)$$

$$\forall h\in\{1,\cdots,n\}\sum_{v=1}^{l}P_{hv}\times\mathrm{Mem}(V_v)\leq\mathrm{Mem}(H_h)\quad(21)$$

节能部署的目标是最大化空闲物理主机的个数 N_{idle}，这些空闲主机可以被关掉以达到节能：

$$N_{\mathrm{idle}}=\sum_{h=1}^{n}x_h,\ x_h=\begin{cases}1, & \sum_{v=1}^{l}P_{hv}=0\\ 0, & \text{其它}\end{cases}\quad(22)$$

根据部署目标的不同，虚拟机部署问题可以表示成不同的形式[41]，下面介绍 3 种不同的虚拟机部署问题：

(1) 开销与性能的权衡(cost performance tradeoff). 部署问题一般要解决两个子问题：① 程序大小的确定；② 程序的部署. 部署的目标是获得最优的性能和开销权衡. 这里的开销只考虑能耗和迁移的开销. 给定一个旧的分配 A_o，一个性能收益函数 $B(A)$，一个能耗开销函数 $P(A)$，一个迁移开销函数 Mig. 对于任何一个分配 A，需要找到一个 A_I(由 x_{ij} 定义，x_{ij} 表示服务器 H_j 上的资源分配给应用程序 V_i)，使得收益最大化：

$$\max\sum_{i=1}^{N}\sum_{j=1}^{M}B(x_{i,j})-\sum_{j=1}^{M}P(A_I)-Mig(A_o,A_I)\quad(23)$$

(2) 性能约束下的成本最小化(cost minimiza-

tion with performance constraint). 目前,数据中心正在走向基于 SLA 的环境,即有量化的性能保证. 因此,在这样一个场景中,性能不再是一个需要最大化的指标,可以被约束条件替换. 在实际操作中,把虚拟机的大小问题从决策器中拿走,虚拟机大小现在由基于 SLA 的性能管理器决定. 决策器只需要最小化分配的整体开销. 因此该优化部署问题变成:

$$\min \sum_{j=1}^{M} P(A_I)+Mig(A_o,A_I) \tag{24}$$

(3)能耗约束下的性能收益最大化(performance benefit maximization with power constraints). 第 3 种程序部署问题是在每个服务器给定一个固定的能耗预算,最大化网络性能收益. 网络性能收益可以通过计算性能收益和迁移开销的差求得

$$\max \sum_{i=1}^{N}\sum_{j=1}^{M} B(x_{i,j})-Mig(A_o,A_I) \tag{25}$$

5.2.2 静态及半静态部署算法

服务器整合可以分为:静态、半静态和动态整合. 静态整合指程序或者虚拟机长时间(如几个月、几年)部署到物理服务器上,不根据负载的变化进行连续迁移. 半静态整合指的是在每天或每周重新整合这些程序. 动态整合要求部署管理器根据运行时特征自动迁移虚拟机,响应负载的变化,时间粒度往往是数个小时.

许多虚拟化软件提供一些工具支持静态整合. 但是这些工具只提供一个基于策略的框架,这些策略需要用户定义,部署的智能化也非常简单. 现在,已经有多个研究提出了动态部署框架. 实际上,管理员往往不愿意自动迁移虚拟机. 他们喜欢离线的或者半离线的框架,部署生效前会先对提出的部署策略进行评估,并经过人工批准. 因此,对于真实数据中心的管理员来说,静态和半静态整合方式(每天或每周进行)是更受欢迎的技术. 通过整合来最小化服务器数目和节能的研究工作较多,但是利用负载的相互关联性信息,系统性地决定最有效的静态整合配置方面的研究工作还较少.

基于负载的特征分析,Verma 等人[35]提出两个新的整合方法:基于相关性的部署(Correlation Based Placement, CBP)和基于峰值聚类的部署(Peak Clustering based Placement, PCP).

(1)基于相关性的部署 CBP. CBP 算法的提出是基于以下一些现象:①一个应用程序使用的峰值资源比其它大多数情况高得多;②如果用非峰值的指标来确定应用程序的规模,并把相关的应用程序部署在一起,会有导致 SLA 冲突的危险;③如果两个不相关的应用程序部署在一起,并且每个应用程序规模冲突概率为 $X\%$,那么在同一时刻,两个应用程序都冲突的概率是$(X\%)^2$.

CBP 算法的基本思路是基于尾部边界(tail bound)来进行应用程序规模的确定,而不是根据应用程序需要的最大规模来判断. 而且 CBP 考虑了程序之间的正相关性约束,这样保证两个正相关的应用程序不会被部署到同一个服务器中. 约束数目可以用可调控的相关性限制进行控制.

同时,因为需要对所有应用程序对的相关性进行计算,CBP 算法也导致了一定的额外开销. 作者证明了给定 N 个应用程序以及有 d 个点的时间序列,CBP 要花费 $O(N^2d)$的时间来找到新的部署.

(2)基于峰值聚类的部署 PCP. PCP 算法的提出是基于以下一些现象:①应用程序峰值的相关性比其它时间段的相关性更重要;②同时出现峰值的一组应用程序,即使以最优的方式被均匀地分布在所有活动主机上,也会出现两个有相关峰值的应用程序被部署在同一个服务器上;③峰值一起出现的共存应用程序可以使用一个共用的缓冲来对付峰值,每个应用程序都预留与非峰值相同的一个值.

PCP 首先把这些有相关峰值的应用程序聚成一类. 但是,如果原始时间序列数据过大,会导致聚类数很多. 因此,PCP 采用一个应用程序原始时间序列两层包围的方法. 用一个值来代表 CPU 利用率分布主体部分,用另一个值来代表其它尾部分布的所有点. 然后,每个活动服务器按应用程序规模为每个聚类进行空间预留,并保留一个与所有聚类中最大峰值同等大的缓冲空间. 每个应用程序根据聚类结果挑选应用程序集合进行预留,然后 PCP 最终为服务器选择应用程序.

5.2.3 动态部署算法

下面介绍 3 种动态部署算法:能耗最小化部署、历史感知部署和迁移开销部署算法[41].

(1)能耗最小化部署算法(Power-minimizing Placement Algorithm),如算法 1 所示,目标是最小化能耗. 首先提出一个 mPP 算法(min Power Parity),把虚拟机部署到给定集合的服务器上,使得所有服务器消耗的能耗最小. 算法分两步:第 1 步,根据服务器的能耗模型决定每个服务器的目标利用率. 目标利用率以贪婪方式进行计算,从 0 利用率开始为

每个服务器计算.然后选择那个单位容量增长能耗最少的服务器.重复这个过程,直到虚拟机都分配完.第 2 步,也叫做 FFD(First Fit Decreasing)装箱算法,是基于 FFD 来部署虚拟机到服务器上,同时满足每个服务器的目标利用率.

算法 1.　能耗最小化部署 mPP.

输入:$\forall i\, VM_i$, $Alloc_{\text{old}}$

输出:$Alloc_{\text{new}}$

$\forall Server_j$

$Alloc_j=\varnothing$, $Used_j=0$

Sort VMs by size in decresing order

FOR $i=1$ TO N

$\forall Server_j$ compute Slope ($Used_j$)

Pick the $Server_{\min}$ with the least Slope

Add VM_i to $Alloc_{\min}$, $Used_{\min}+=Size(VM_i)$

END FOR

$Alloc_{\text{new}}=$FFD ($Used$)

Return $Alloc_{\text{new}}$

(2)历史感知的部署算法(History Aware Packing Algorithm),如算法 2 所示,目标是减少迁移次数.

算法 2.　历史感知的部署 iFFD.

输入:$Alloc_0$, $Used$

输出:$Alloc_n$

$Donors=\varnothing$, $Receivers=\varnothing$

FOR all servers S_j

$Prev_j=$sum of VMs in S_j by $Alloc_0$

IF ($Prev_j>Used_j$)

Add S_j to $Donors$

$Mig_j=Prev_j-Used_j$

ELSE

Add S_j to $Receivers$

END FOR

FOR all S_j in $Donors$

Pick the smallest VMs that add upto

Mig_j and add them to $MigList$

END FOR

Sort $MigList$ based on size

FOR all VM_i in $MigList$

Place VM_i on the first $Donor_j$ that can pack it within

$Used_j$

END FOR

Return $Alloc_n$

能量最小化部署算法 mPP 的目标是最小化能耗,但是算法没有考虑也不知道前一次的配置结果.因此可能导致大规模的迁移操作,这会导致很高的整体开销(能耗和迁移).因此,Verma 等人提出了一个改进的 FFD 算法,称做 iFFD(incremental FFD)来进行应用程序到物理服务器上的部署.

iFFD 首先计算需要请求更高利用率的服务器列表,用 receivers 标记.对于每个 donor(目标利用率低于当前的利用率),选择规模最小的应用程序来迁移,把它们添加到一个虚拟机迁移列表.然后运行 FFD,把 receivers 剩余容量(目标容量-当前容量)作为箱子,虚拟机迁移列表作为球,进行装箱操作.

(3)迁移开销感知的部署算法(Migration Cost-aware Locally Optimal Placement Algorithm),如算法 3 所示,目标是最小化总开销,包括能耗和迁移的开销,也就是考虑单位迁移开销的能耗优化.

算法 3.　迁移开销感知的局部优化部署 pMaP.

输入:$Alloc_0$, VM_i

输出:$Migs$

$Alloc_n=$mPPH ($Alloc_0$, VM_i)

$MList=$getMigList($Alloc_0$, $Alloc_n$)

$\forall Server_j$ with no VMs placed in $Alloc_n$

$VG_j=$VMs placed on $Server_j$ in $Alloc_0$

Add VG_j to $MList$

$\forall mig_i\in MList$

$Cost_i=getMigrationCost(mig_i)$,

$Benefit_i=getBenefit(mig_i)$

Sort $MList$ by $Benefit_i/cost_i(decreasing)$

$mig_{\text{best}}=$most profitable entry in $MList$

WHILE ($profit_{\text{best}}>cost_{\text{best}}$) AND ($MList\neq\varnothing$)

$Migs=Migs\cup mig_{\text{best}}$

Delete mig_{best} from $MList$

Recompute Cost and Benefit for $MList$

END WHILE

Return $Migs$

pMaP 算法平衡能耗和迁移的开销,目标是寻找一个最小化总开销(能耗和迁移)的分配.pMaP 以最小化能耗的方式连续寻找新的虚拟机分配,同时考虑迁移的开销.算法首先调用任意的能耗最小化部署算法,得到一个新的能耗最小化部署,然后比较两种部署的差异,决定选择一个子集.选择过程是基于对所有迁移的单位迁移开销的能耗上升量进行升序排序.然后,选择收益最大的迁移.重复以上步骤,直到没有能耗和迁移权衡优化的迁移存在.

此外,Moore 等人[80]开发了温度感知的负载部署算法,最大限度减少为冷却基础设施而支出的能耗,实现较低的冷却开销,增加硬件可靠性.他们通过观察数据中心中热气流的方法,提出两种负载部署策略:基于区域离散化(Zone-Based Discretiza-

tion,ZBD)和最小化热循环(Minimize-Heat-Recirculation, MinHR). Verma 等人[81]特别研究了在虚拟化支持的服务器上,对高性能计算应用程序使用节能管理技术,提出了一个框架和方法论进行功耗感知的高性能计算程序部署. Lim 等人[53]采用虚拟机在线迁移技术,提出了一种功耗感知的域部署(PADD)策略.

5.3 节能整合算法

Lee 和 Zomaya[5]提出了两种能耗感知的任务整合算法:ECTC 和 MaxUtil. ECTC 和 MaxUtil 的步骤相似(如算法 4 所示),主要的区别在于它们的开销函数. 对于一个给定的任务,这两个启发式算法检测每个资源,把最高能效的资源分配给该任务.

算法 4. 能耗感知的任务整合算法.

输入: A task t_j and a set R of r cloud resources
输出: A task-resource match
Let $r^*=\varnothing$
FOR $\forall r_i \in R$ DO
  Compute the cost function value $f_{i,j}$ of t_j on r_i
  IF $f_{i,j} > f_{*j}$ THEN
    Let $r^*=r_i$
    Let $f_{*j}=f_{i,j}$
  END IF
END FOR
Assgin t_j to r^*

ECTC 的开销函数计算当前任务的真实能耗减去运行一个任务的最小能耗 $P_{\min}$,如果有其它任务跟这个任务并行运行的话,那么这些任务和当前任务在重叠时间的能量消耗需要被考虑进来. 开销函数往往区别对待单独运行的任务. 使用 ECTC 开销函数,任务 t_j 在资源 r_i 获得的值 $f_{i,j}$ 被定义为

$$f_{i,j}=((p_\Delta \times u_j + p_{\min}) \times \tau_0) - ((p_\Delta \times u_j + p_{\min}) \times \tau_1 + p_\Delta \times u_j \times \tau_2) \quad (26)$$

其中,p_Δ 是 $p_{\max}$ 和 $p_{\min}$ 的差值,u_j 是 t_j 的利用率,τ_0、τ_1、τ_2 分别是 t_j 的总体处理时间、单独运行时间和并行运行时间. 该函数表明最低利用率时的能耗比空闲时的能耗多的多,由重叠任务导致的能耗增加幅度相对较少.

MaxUtil 的开销函数由在当前任务的处理时间内的主要系统部件的平均利用率推断而来. 该函数的目标是增加整合的密度. 有两方面的优点:第 1 个优点是降低了能量消耗;第 2 个优点是 MaxUtil 的开销函数暗中降低了活动资源的个数. 因为与 ECTC 的开销函数相比的话,该开销函数趋向于少数高利用率的资源. 使用 MaxUtil 开销函数,任务 t_j 在资源 r_i 获得的值 $f_{i,j}$ 被定义为

$$f_{i,j}=\frac{\sum_{\tau=1}^{\tau_0} U_i}{\tau_0} \quad (27)$$

此外,Zhu 等人[52]针对虚拟化环境下的科学工作流,开发了一个能耗感知的整合框架 pSciMapper. Beloglazov 等人[68]基于虚拟机资源使用的历史数据的分析,提出了一种新的自适应启发式算法进行虚拟机的动态整合,该算法在 1000 台虚拟机上进行了验证分析,实验证明算法能大幅减少能耗,同时保证较高的 SLA. Berral 等人[82]从机器学习技术角度,研究虚拟机的动态整合问题,同时满足能耗和 SLA 的双重目标.

5.4 节能迁移算法

Beloglazov 等人[47]提出了 4 种启发式的算法来选择虚拟机进行迁移. 其中单阈值方法(Single Threshold,ST),是基于设置主机上限利用率的想法,部署虚拟机的时候保持总 CPU 利用率低于这个阈值. 目标是预留空闲资源,防止在整合场景下虚拟机资源需求增加而导致的 SLA 冲突. 在每个时间段,所有的虚拟机根据 MBFD(Modified Best Fit Decreasing)算法进行重分配,额外的条件是保持上限利用率阈值没有超过. 新的部署可以通过虚拟机在线迁移来进行.

其它 3 个启发式算法是基于设置主机利用率上下限,保持整体 CPU 利用率在这两个阈值之间. 如果主机 CPU 的利用率低于下界,所有的虚拟机需要被迁移走,这个主机需要被关掉,消除空闲主机的能耗. 如果利用率高于上界,则一些虚拟机需要被迁移走以降低利用率,防止潜在 SLA 冲突. 作者提出了 3 种策略来选择需要从主机迁移走的虚拟机:(1)最小化迁移(Minimization of Migrations, MM). 迁移最小的虚拟机个数来最小化迁移开销;(2)最高增长潜力(Highest Potential Growth, HPG). 迁移那些有最低 CPU 使用率的虚拟机,为了最小化整体潜在的利用率上升,进而导致 SLA 冲突;(3)随机选择(Random Choice,RC). 迁移必要数目的虚拟机,根据一个均匀分布的随机变量进行选择.

最近,Graubner 等人[83]提出了一个提升 IaaS 云的能效的方法. 与前人工作不同的地方在于,该工作考虑了在线迁移过程中的预处理和后处理阶段的

能量开销，并在 Eucalyptus 开源云计算系统中实现.

5.5 其它算法

Mazzucco 等人[90]提出了一个能量感知的分配策略，目标是最大化平均收益. 这个策略基于动态评估用户需求和建模系统行为. Le 等人[91]提出并评估了一个基于优化的请求分布框架，进行多数据中心的分配. 这个框架允许服务管理它们自己的能耗和开销，同时满足 SLA. 基于这个框架，提出两种请求分配策略：考虑不同的时间区域和不同的电力价格；考虑绿色能源.

Srikantaiah 等人[28]把整合问题建模成一个改进的装箱问题，考虑 CPU 和磁盘使用率. 该算法尝试在最优点整合任务，平衡能耗和性能. 算法分两步：(1) 从轮廓分析数据决定最优点；(2) 使用每个服务器当前分配和最优分配之间的欧氏距离，进行能量感知的资源分配.

Liao 等人[44]提出了一个概率启发式算法来优化在多维资源约束下的虚拟机在物理机上的映射问题，并给出了一个经济学方法来平衡能耗降低和性能损耗的问题.

Rodero 等人[92]针对高性能计算程序，提出了一个在整合的虚拟化计算平台中的能量感知的在线供给算法. 通过使用一个负载感知的动态供给机制，以及关闭空闲主机等操作来获得节能. 对新进来的任务请求，根据他们的系统配置和运行时状态，使用一个在线的聚类方法来动态特征化和聚类，再根据详细的虚拟机配置需求聚类结果来确定节能的资源供给.

6 结　语

本文分析了虚拟化云计算平台的能耗管理技术，从能耗测量、能耗建模、能耗管理机制、能耗管理算法 4 个方面进行了系统的阐述. 能耗测量的开销和准确性是制定高效能耗管理策略的基础. 能耗建模的精度，直接反映了虚拟机能耗数据的准确性，对后续节能优化非常重要. 能耗管理机制被分为虚拟化层的管理机制和云平台层的管理机制. 能耗管理算法按应用场景被分为节能部署算法、节能整合算法、节能迁移算法等. 这 4 方面的内容关系密切，相互衔接，共同构成了虚拟化云计算平台的能耗管理解决方案. 它们的关系是：能耗测量获得资源使用情况的原始数据，传递给能耗模型；能耗模型根据计算得出虚拟机的能耗使用情况；基于这些数据，可以实现复杂的能耗管理机制和管理算法.

虚拟化云计算平台的能耗管理目前还处在刚刚兴起的阶段，大多数研究成果是最近几年才发表的，还存在很多问题需要进一步研究. 下面我们根据对前人工作的总结，并结合自己的理解，给出未来虚拟化云计算平台能耗管理领域的十个需进一步研究的问题：

(1) 多因素驱动的轻量级在线能耗轮廓分析方法. 能耗问题不是一个孤立的问题，它受到多方面因素的影响. 在真实的云计算环境中，往往需要一种在线的能耗监测方法能及时准确地提供虚拟机的能耗使用数据，但同时又希望能耗监测的过程是轻量的，即占用少量的系统资源即可完成.

(2) 基于统计学习的能耗建模技术. 由于能耗相关的硬件事件种类和参数众多，需要引入基于统计学习的方法对原始监测数据进行统计分析，不断修正能耗模型，提升模型的精度. 同时这种基于历史数据学习的建模方法，还可以用来预测未来的能量消耗情况，提前进行相应的节能决策.

(3) 能耗和性能的双目标优化. 一个有效的节能方法不仅仅能减少能耗，而且要保证不牺牲性能. 由于有时性能(包括 QoS 或 SLA)和能耗的优化是互斥的，例如服务器整合能减少物理主机数目，达到节能目的，但是服务器整合同时会因共享资源的竞争和干扰影响共存应用程序的性能，因此需要一种双目标的优化方法.

(4) 迁移开销和状态切换开销的分析. 在线迁移技术已经成为实现数据中心层次节能的一个重要技术. 但迁移不是免费的，它本身是存在开销，包括网络传输的开销以及存在短暂的停机时间(往往几十毫秒). 另外，服务器在休眠和唤醒之间进行状态的切换也是需要开销的. 很多前人的工作都忽略了这一点，因此需要进行重新考虑.

(5) 多维资源的节能. 目前的大多数的工作都只考虑计算资源的节能，较少涉及其它诸如存储资源(包括文件系统)、网络资源等的节能. 当计算资源的节能优化已经发展相对成熟的时候，多关注其它系统资源的节能或许可以获得良好的效果.

(6) 能耗感知的资源管理算法研究. 传统的资源管理方法主要关注性能的提升，当考虑能耗因素时，这些方法可能会发生变化. 研究能耗感知的云计

算资源管理算法,包括节能调度、节能分配、节能部署等,能有效降低云数据中心的能耗.

(7)个性化节能方案的推荐.在云计算环境中,负载是多种多样的,它们对资源的需求和对服务器能效的需求也是不一样的,同时系统的软硬件运行环境也是存在差异的.如何根据已有的信息,主动地推荐个性化的节能方案是一个有意思的问题.

(8)复杂应用程序的能耗分析.通用的节能方法(如整合)对于不同的应用程序来说,获得的效果参差不齐.因此,需要分析特定应用程序(如 Multi-tier 程序、数据密集程序等)的行为特征和资源访问模式,进行针对性的优化.

(9)协同节能方法.当前,不同的系统部件和层次,如处理器层、操作系统层、集群层、数据中心层等,都提供各自的能耗管理机制.此外不同的软硬件制造商也会提供各自的能耗管理机制.如何协同这些处在不同层次的、来源不同的能耗管理机制,发挥最大的节能优势,是一个需要解决的问题.

(10)虚拟化能耗的标准化评价问题.针对传统服务器的能耗测试基准有 SPECpower_ssj2008 和 TPC_Energy,还没有专门的针对虚拟化云计算环境的能耗测试基准.因此,需要研究虚拟化云计算环境下的能耗评价面临的新问题,定义评价指标并制定评价方法.

参 考 文 献

[1] Armbrust M, Fox A, Griffith R, Joseph A D, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I et al. A view of cloud computing. Communications of the ACM, 2010, 53(4): 50-58

[2] Chen Kang, Zheng Wei-Min. Cloud computing: System instances and current research. Journal of Software, 2009, 20(5): 1337-1348(in Chinese)
(陈康,郑纬民. 云计算:系统实例与研究现状. 软件学报, 2009, 20(5): 1337-1348)

[3] Hooper A. Green computing. Communications of the ACM, 2008, 51(10): 11-13

[4] Ranganathan P. Recipe for efficiency: Principles of power-aware computing. Communications of the ACM, 2010, 53(4): 60-67

[5] Lee Y C, Zomaya A Y. Energy efficient utilization of resources in cloud computing systems. The Journal of Supercomputing, 2010, 1-13

[6] Bohrer P, Elnozahy E N, Keller T, Kistler M, Lefurgy C, McDowell C, Rajamony R. The case for power management in web servers//Graybill Robert; Melhem Rami eds. Power Aware Computing. Germany: Springer, 2002: 1-31

[7] Barroso L A, Holzle U. The case for energy-proportional computing. IEEE Computer, 2007, 40(12): 33-37

[8] Ranganathan P, Leech P, Irwin D, Chase J. Ensemble-level power management for dense blade servers//Proceedings of the 33rd Annual International Symposium on Computer Architecture(ISCA'06). Boston, USA, 2006: 66-77

[9] Barham P, Dragovic B, Fraser K, Hand S, Harris T, Ho A, Neugebauer R, Pratt I, Warfield A. Xen and the art of virtualization//Proceedings of the 19th ACM Symposium on Operating Systems Principles(SOSP'03). New York, USA, 2003: 164-177

[10] Hermenier F, Lorca X, Menaud J M, Muller G, Lawall J. Entropy: A consolidation manager for clusters//Proceedings of the 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments(VEE'09). Washington, USA, 2009: 41-50

[11] Clark C, Fraser K, Hand S, Hansen J G, Jul E, Limpach C, Pratt I, Warfield A. Live migration of virtual machines//Proceedings of the 2nd Conference on Symposium on Networked Systems Design & Implementation (NSDI'05). Boston, USA, 2005: 273-286

[12] Ye K, Jiang X, Ye D, Huang D. Two optimization mechanisms to improve the isolation property of server consolidation in virtualized multi-core server//Proceedings of the 12th IEEE International Conference on High Performance Computing and Communications(HPCC'10). Melbourne, Australia, 2010: 281-288

[13] Cully B, Lefebvre G, Meyer D, Feeley M, Hutchinson N, Warfield A. Remus: High availability via asynchronous virtual machine replication//Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation(NSDI'05). Boston, USA, 2008: 161-174

[14] Ye K, Jiang X, He Q, Li X, Chen J. Evaluate the performance and scalability of image deployment in virtual data center//Proceedings of the 2010 IFIP International Conference on Network and Parallel Computing. Zhengzhou, China, 2010: 390-401

[15] Patterson A D. The data center is the computer. Communications of the ACM, 2008, 51(1): 105-105

[16] Buyya R, Beloglazov A, Abawajy J. Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges//Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'10). Las Vagas, USA, 2010: 6-20.

[17] Guo Bing, Shen Yan, Shao Zi-Li. The redefinition and some discussion of green computing. Chinese Journal of Computers, 2009, 32(12): 2311-2319(in Chinese)
(郭兵,沈艳,邵子立. 绿色计算的重定义与若干探讨. 计算机学报, 2009, 32(12): 2311-2319)

[18] Berl A, Gelenbe E, Di Girolamo M, Giuliani G, De Meer H, Dang M Q, Pentikousis K. Energy-efficient cloud computing. The Computer Journal, 2010, 53(7): 1045-1051

[19] Nathuji R, Schwan K. VirtualPower: Coordinated power management in virtualized enterprise systems//Proceedings of the 21th ACM SIGOPS Symposium on Operating Systems Principles (SOSP'07). Washington, USA, 2007: 265-278

[20] Nathuji R, Schwan K, Somani A, Joshi Y. VPM tokens: Virtual machine-aware power budgeting in datacenters. Cluster Computing, 2009, 12(2): 189-203

[21] Stoess J, Lang C, Bellosa F. Energy management for hypervisor-based virtual machines//Proceedings of the USENIX Annual Technical Conference (USENIX ATC'07). Santa Clara, USA, 2007: 1-14

[22] Soundararajan V, Anderson J M. The impact of management operations on the virtualized datacenter//Proceedings of the ACM International Symposium on Computer Architecture (ISCA'10). Saint-Malo, France, 2010: 326-337

[23] Kansal A, Zhao F, Liu J, Kothari N, Bhattacharya A A. Virtual machine power metering and provisioning//Proceedings of the 1st ACM Symposium on Cloud Computing. Indianapolis, USA, 2010: 39-50

[24] McIntosh-Smith S, Wilson T, Crisp J, Ibarra A, Sessions R B. Energy-aware metrics for benchmarking heterogeneous systems. ACM SIGMETRICS Performance Evaluation Review, 2011, 38(4): 88-94

[25] Krishnan B, Amur H, Gavrilovska A, Schwan K. VM power metering: Feasibility and challenges. ACM SIGMETRICS Performance Evaluation Review, 2011, 38(3): 56-60

[26] Liu L, Wang H, Liu X, Jin X, He W B, Wang Q B, Chen Y. GreenCloud: A new architecture for green data center//Proceedings of the 6th International Conference on Autonomic Computing and Communications (ICAC'09). Barcelona, Spain, 2009: 29-38

[27] Bohra A E H, Chaudhary V. VMeter: Power modelling for virtualized clouds//Proceedings of the 2010 IEEE International Symposium on Parallel & Distributed (IPDPS'10). Atlanta, USA, 2010: 1-8

[28] Srikantaiah S, Kansal A, Zhao F. Energy aware consolidation for cloud computing//Proceedings of the 2008 Conference on Power Aware Computing and Systems (HotPower'08). San Diego, USA, 2008: 1-5

[29] Kansal A, Zhao F. Fine-grained energy profiling for power-aware application design. ACM SIGMETRICS Performance Evaluation Review, 2008, 36(2): 26-31

[30] Choi J, Govindan S, Urgaonkar B, Sivasubramaniam A. Profiling, prediction, and capping of power consumption in consolidated environments//Proceedings of the IEEE International Symposium on Modeling, Analysis and Simulation of Computers and Telecommunication Systems (MASCOTS'08). Baltimore, USA, 2008: 1-10

[31] Choi J, Govindan S, Jeong J, Urgaonkar B, Sivasubramaniam A. Power consumption prediction and power-aware packing in consolidated environments. IEEE Transactions on Computers, 2010, 59(12): 1640-1654

[32] Lin Chuang, Tian Yuan, Yao Min. Green network and green evaluation: Mechanism, modeling and evaluation. Chinese Journal of Computers, 2011, 34(4): 593-612(in Chinese)
(林闯, 田源, 姚敏. 绿色网络和绿色评价: 节能机制、模型和评价. 计算机学报, 2011, 34(4): 593-612)

[33] Bao Y, Chen M, Ruan Y, Liu L, Fan J, Yuan Q, Song B, Xu J. HMTT: A platform independent full-system memory trace monitoring system. ACM SIGMETRICS Performance Evaluation Review, 2008, 36(1): 229-240

[34] Apparao P, Iyer R, Zhang X, Newell D, Adelmeyer T. Characterization & analysis of a server consolidation benchmark//Proceedings of the 4th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE'08). Seattle, USA, 2008: 21-30

[35] Verma A, Dasgupta G, Nayak T K, De P, Kothari R. Server workload analysis for power minimization using consolidation//Proceedings of the 2009 Conference on USENIX Annual Technical Conference (USENIX ATC'09). San Diego, USA, 2009: 28-41

[36] Ye K, Jiang X, Huang D, Chen J, Wang B. Live migration of multiple virtual machines with resource reservation in cloud computing environments//Proceedings of the 4th IEEE International Conference on Cloud Computing (Cloud'11). Washington, USA, 2011: 267-274

[37] Zhang Bin-Bin, Luo Ying-Wei, Wang Xiao-Lin, Wang Zhen-Lin, Sun Yi-Feng, Chen Hao-Gang, Xu Zhuo-Qun, Li Xiao-Ming. Whole-system live migration mechanism for virtual machines. Chinese Journal of Electronics, 2009, 37(4): 894-899 (in Chinese)
(张彬彬, 罗英伟, 汪小林, 王振林, 孙逸峰, 陈昊罡, 许卓群, 李晓明. 虚拟机全系统在线迁移. 电子学报, 2009, 37(4): 894-899)

[38] Liu H, Xu C Z, Jin H, Gong J, Liao X. Performance and energy modeling for live migration of virtual machines//Proceedings of the 20th International Symposium on High Performance Distributed Computing (HPDC'11). San Jose, USA, 2011: 171-182

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

[40] Jung G, Hiltunen M A, Joshi K R, Schlichting R D, Pu C. Mistral: Dynamically managing power, performance, and adaptation cost in cloud infrastructures//Proceedings of the 2010 International Conference on Distributed Computing Systems (ICDCS'10). Genoa, Italy, 2010: 62-73

[41] Verma A, Ahuja P, Neogi A. pMapper: Power and migration cost aware application placement in virtualized systems//Proceedings of the 9th ACM/IFIP/USENIX Middleware Conference (Middleware'08). Leuven, Belgium, 2008: 243-264

[42] Hu L, Jin H, Liao X, Xiong X, Liu H. Magnet: A novel scheduling policy for power reduction in cluster with virtual machines//Proceedings of the 2008 IEEE International Conference on Cluster Computing (Cluster'08). Tsukuba, Japan, 2008: 13-22

[43] Chen H, Jin H, Shao Z, Hu K, Yu K, Tian K. ClientVisor: Leverage COTS OS functionalities for power management in virtualized desktop environment. ACM SIGOPS Operating Systems Review, 2009, 43(3): 62-71

[44] Liao X, Jin H, Liu H. Towards a green cluster through dynamic remapping of virtual machines. Future Generation Computer Systems, 2012, 28(2): 469-477

[45] Ye K, Huang D, Jiang X, Chen H, Wu S. Virtual machine based energy-efficient data center architecture for cloud computing: A performance perspective//Proceedings of the 2010 IEEE/ACM International Conference on Green Computing and Communications (GreenCom'10). Hangzhou, China, 2010: 171-178

[46] Shi Y, Jiang X, Ye K. An energy-efficient scheme for cloud resource provisioning based on CloudSim//Proceedings of the 2011 IEEE International Conference on Cluster Computing (Cluster'11). Austin, USA, 2011: 595-599

[47] Beloglazov A, Buyya R. Energy efficient allocation of virtual machines in cloud data centers//Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid'10). Melbourne, Australia, 2010: 577-578

[48] Jang J W, Jeon M, Kim H S, Jo H, Kim J S, Maeng S. Energy reduction in consolidated servers through memory-aware virtual machine scheduling. IEEE Transactions on Computers, 2011, 99(1): 552-564

[49] Kaushik R, Bhandarkar M. GreenHDFS: Towards an energy-conserving storage-efficient, hybrid hadoop compute cluster//Proceedings of the USENIX Workshop on Power Aware Computing and Systems (HotPower'10). Vancouver, Canada, 2010: 1-5

[50] Seetharaman S. Energy conservation in multi-tenant networks through power virtualization//Proceedings of the 2010 International Conference on Power Aware Computing and Systems (HotPower'10). Vancouver, Canada, 2010: 1-8

[51] Das T, Padala P, Padmanabhan V N, Ramjee R, Shin K G. Litegreen: Saving energy in networked desktops using virtualization//Proceedings of the 2010 USENIX Conference on USENIX Annual Technical Conference (USENIX ATC'10). Boston, USA, 2010: 1-15

[52] Zhu Q, Zhu J, Agrawal G. Power-aware consolidation of scientific workflows in virtualized environments//Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC'10). New Orleans, USA, 2010: 1-12

[53] Lim M Y, Rawson F, Bletsch T, Freeh V W. PADD: Power aware domain distribution//Proceedings of the 29th IEEE International Conference on Distributed Computing Systems (ICDCS'09). Montreal, Canada, 2009: 239-247

[54] Ye L, Lu G, Kumar S, Gniady C, Hartman J H. Energy-efficient storage in virtual machine environments//Proceedings of the 6th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE'10). Pittsburgh, USA, 2010: 75-84

[55] Beloglazov A, Buyya R, Lee Y C, Zomaya A. A taxonomy and survey of energy-efficient data centers and cloud computing systems. Advances in Computers, 2011, 82(2): 47-111

[56] Wei G, Liu J, Xu J, Lu G, Yu K, Tian K. The on-going evolutions of power management in Xen. Intel Corporation Tech. Rep., 2009

[57] Oh Y F, Kim S H, Eom H, Yeom Y H. Enabling consolidation and scaling down to provide power management for cloud computing//Proceedings of the 3rd USENIX Workshop on Hot Topics in Cloud Computing (HotCloud'11). Portland, USA, 2011: 1-5

[58] Dhiman G, Marchetti G, Rosing T. vGreen: A system for energy efficient computing in virtualized environments//Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED'09). San Francisco, USA, 2009: 243-248

[59] Van H N, Tran F D, Menaud J M. Performance and power management for cloud infrastructures//Proceedings of the 3rd International Conference on Cloud Computing (Cloud'10). Miami, USA, 2010: 329-336

[60] Ye K, Che J, Jiang X, Chen J, Li X. vTestkit: A performance benchmarking framework for virtualization environments//Proceedings of the 5th Annual ChinaGrid Conference. Guangzhou, China, 2010: 130-136

[61] Ye K, Jiang X, Chen S, Huang D, Wang B. Analyzing and modeling the performance in Xen-based virtual cluster environment//Proceedings of the 12th IEEE International Conference on High Performance Computing and Communications (HPCC'10). Melbourne, Australia, 2010: 273-280

[62] Lin Chuang, Li Yin, Wan Jian-Xiong. Optimization approaches for QoS in computer networks: A survey. Chinese Journal of Computers, 2011, 34(1): 1-14(in Chinese)
(林闯, 李寅, 万剑雄. 计算机网络服务质量优化方法研究综述. 计算机学报, 2011, 34(1): 1-14)

[63] Dyachuk D, Mazzucco M. On allocation policies for power and performance//Proceedings of the 11th IEEE/ACM International Conference on Grid Computing (Grid'10). Brussels, Belgium, 2010: 313-320

[64] Chase J S, Anderson D C, Thakar P N, Vahdat A M, Doyle R P. Managing energy and server resources in hosting centers. ACM SIGOPS Operating Systems Review, 2001, 35(5): 103-116

[65] Kephart J O, Chan H, Das R, Levine D. W, Tesauro G, Rawson F, Lefurgy C. Coordinating multiple autonomic managers to achieve specified power-performance tradeoffs//

Proceedings of the 4th International Conference on Autonomic Computing (ICAC'07). Jacksonville, USA, 2007: 1-10

[66] Park S, Jiang W, Zhou Y, Adve S. Managing energy-performance tradeoffs for multithreaded applications on multiprocessor architectures. ACM SIGMETRICS Performance Evaluation Review, 2007, 35(1): 169-180

[67] Gandhi A, Gupta V, Harchol-Balter M, Kozuch M A. Optimality analysis of energy-performance trade-off for server farm management. Performance Evaluation, 2010, 67(11): 1155-1171

[68] Beloglazov A, Buyya R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concurrency and Computation: Practice and Experience, 2011, doi: 10.1002/cpe.1867

[69] Albers S. Energy-efficient algorithms. Communications of the ACM, 2010, 53(5): 86-96

[70] Bradley D J, Harper R E, Hunter S W. Workload-based power management for parallel computer systems. IBM Journal of Research and Development, 2003, 47(5): 703-718

[71] Chen G, He W, Liu J, Nath S, Rigas L, Xiao L, Zhao F. Energy-aware server provisioning and load dispatching for connection-intensive internet services//Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation (NSDI'08). San Francisco, USA, 2008: 337-350

[72] Kusic D, Kephart J O, Hanson J E, Kandasamy N, Jiang G. Power and performance management of virtualized computing environments via lookahead control. Cluster Computing, 2009, 12(1): 1-15

[73] Wang X, Wang Y. Coordinating power control and performance management for virtualized server clusters. IEEE Transactions on Parallel and Distributed Systems, 2011, 22(2): 245-259

[74] Wang X, Wang Y. Co-Con: Coordinated control of power and application performance for virtualized server clusters//Proceedings of the 17th International Workshop on Quality of Service (IWQoS'09). Charleston, USA, 2009: 1-9

[75] Wang Y, Wang X, Chen M, Zhu X. Partic: Power-aware response time control for virtualized web servers. IEEE Transactions on Parallel and Distributed Systems, 2011, 22(2): 323-336

[76] Raghavendra R, Ranganathan P, Talwar V, Wang Z, Zhu X. No "power" struggles: Coordinated multi-level power management for data center//Proceedings of the 13th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS XIII). Seattle, USA, 2008: 48-59

[77] Rajamani K, Lefurgy C. On evaluating request-distribution schemes for saving energy in server clusters//Proceedings of the IEEE Symposium on Performance Analysis of Systems and Software (ISPASS'03). Austin, USA, 2003: 111-122

[78] Ghiasi S, Felter W. CPU packing for multiprocessor power reduction//Falsafi B, Vijaykumar T N eds. Power-Aware Computer Systems. Berlin: Springer-Verlag, 2003: 117-131

[79] Banikazemi M, Poff D, Abali B. PAM: A novel performance/power aware meta-scheduler for multi-core systems//Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'08). Austin, USA, 2008: 1-12

[80] Moore J, Chase J, Ranganathan P, Sharma R. Making scheduling cool: Temperature-aware workload placement in data centers//Proceedings of the Annual Conference on USENIX Annual Technical Conference (USENIX ATX'05). Anaheim, USA, 2005: 61-74

[81] Verma A, Ahuja P, Neogi A. Power-aware dynamic placement of HPC applications//Proceedings of the 22nd Annual International Conference on Supercomputing (SC'08). Austin, USA, 2008: 175-184

[82] Berral J L, Goiri í, Nou R, Julià F, Guitart J, Gavaldà R, Torres J. Towards energy-aware scheduling in data centers using machine learning//Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking (e-Energy'10). Passau, Germany, 2010: 215-224

[83] Graubner P, Schmidt M, Freisleben B. Energy-efficient management of virtual machines in Eucalyptus//Proceedings of the 2011 IEEE International Conference on Cloud Computing (CLOUD'11). Washington, USA, 2011: 243-250

[84] Lefurgy C, Wang X, Ware M. Power capping: A prelude to power shifting. Cluster Computing, 2008, 11(2): 183-195

[85] Wu Q, Juang P, Martonosi M, Peh L S, Clark D W. Formal control techniques for power-performance management. IEEE Micro, 2005, 25(5): 52-62

[86] Wang Y, Wang X, Chen M, Zhu X. Power-efficient response time guarantees for virtualized enterprise servers//Proceedings of the 2008 Real-Time Systems Symposium (RTSS'08). Barcelona, Spain, 2008: 303-312

[87] Padala P, Shin K G, Zhu X, Uysal M, Wang Z, Singhal S, Merchant A, Salem K. Adaptive control of virtualized resources in utility computing environments. ACM SIGOPS Operating Systems Review, 2007, 41(3): 289-302

[88] Zhang Y, Bestavros A, Guirguis M, Matta I, West R. Friendly virtual machines: Leveraging a feedback-control model for application//Proceedings of the 1st ACM/USENIX International Conference on Virtual Execution Environments (VEE'05). Chicago, USA, 2005: 2-12

[89] Dasgupta G, Sharma A, Verma A, Neogi A, Kothari R. Workload management for power efficiency in virtualized data-centers. Communications of the ACM, 2011, 54(7): 131-141

[90] Mazzucco M, Dyachuk D, Deters R. Maximizing cloud providers' revenues via energy aware allocation policies//Proceedings of the 3rd International Conference on Cloud Computing (Cloud'10). Washington, USA, 2010: 131-138

[91] Le K, Bianchini R, Martonosi M, Nguyen T. Cost-and energy-aware load distribution across data centers//Proceedings of the USENIX Workshop on Power Aware Computing and Systems(HotPower'09). Montana, USA, 2009: 1-5

[92] Rodero I, Jaramillo J, Quiroz A, Parashar M, Guim F. Towards energy-aware autonomic provisioning for virtualized environments//Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (HPDC'10). Chicago, USA, 2010: 320-323

附 录.

附表 1 虚拟化云计算平台能耗管理机制的综合比较

机制名称	实现平台	资源维度	同构/异构	是否迁移	优化目标	具体技术	实施效果
Energy Efficient Resource Utilization[5]	仿真环境	CPU	N/A	N/A	最小化能耗,维持性能	两个能耗感知的整合方法	可以有效节能
VirtualPower[19]	Xen	CPU	异构	迁移	最小化能耗,满足性能约束	硬伸缩,软伸缩,整合/迁移	节能 34%
VPM Token[20]	Xen	CPU	异构	迁移	维持能耗约束	VM 感知的能耗预算	能耗约束下降 43%,效用提升
Hypervisor-Based Energy Management[21]	L4	CPU,磁盘	同构	N/A	实现能耗感知的记账和分配	预算分配	可以有效节能
Workload Analysis based Power Minimization[35]	IBM Emerald	CPU	异构	N/A	最大化节能效果,同时维持性能	静态与半静态整合	可以有效节能
Mistral[40]	Xen	N/A	N/A	N/A	最大化总体效用,优化能耗和性能	多层自适应分析及优化算法	提升了整体效用
pMapper[41]	VMware ESX Hypervisor	CPU	异构	迁移	最小化能耗,最小化性能损失	整合,服务器功率切换	可以有效节能
Magnet[42]	Xen	CPU、内存、I/O	同构/异构	迁移	降低能耗,保持性能	迁移	节能 74.8%
ClientVisor[43]	Xen	CPU 和 I/O	同构	N/A	最小化能耗	采用操作系统提供的节能功能	节能 22%
GreenMap[44]	Xen	多维资源	同构	N/A	节能能耗,维持性能	运行时 VM 映射框架	节能 69.2%
Energy-Efficient Architecture[45]	Xen/KVM	N/A	同构	迁移	最小化能耗,满足性能约束	整合,迁移	可以有效节能
Energy-Efficient Provisioning[46]	CloudSim	CPU	同构	N/A	最小化能耗,满足性能约束	DVFS,预测方法	节能 10.98%
Energy Efficient Allocation[47]	CloudSim	CPU	N/A	迁移	最小化能耗,保证 QoS	动态虚拟机重分配	节能 83%
Energy Efficient Scheduling[48]	Xen,MPSim	内存	N/A	N/A	减少内存的能耗	内存感知的 VM 调度	节能 57.4%
GreenHDFS[49]	Hadoop	文件系统	同构	N/A	降低系统能耗	Cold zone 能耗管理	节能 26%
Energy Efficient Network[50]	N/A	网络设备	N/A	N/A	降低网络设备能耗	整合	可以有效节能
LiteGreen[51]	MS Hyper-V	CPU,网络	同构	迁移	节约桌面能量,最小化用户冲突	迁移	节能 74%
pSciMapper[52]	Xen	科学工作流	N/A	N/A	降低能耗和资源开销,同时维持高的性能	整合	节能 56%
PADD[53]	仿真环境	CPU	同构	迁移	最小化整体能耗,同时避免 SLA 冲突	迁移、自适应缓存	节能 70%
Energy-Efficient Storage[54]	Xen	磁盘	N/A	N/A	减少磁盘的能量消耗	VMM 缓冲	节能 14.8%

注:N/A 表示论文中没有提及.

附 表 2　虚拟化云计算平台的能耗管理算法比较

分类依据	分类类别	说明	优点	缺点	典型案例
按主被动模式分	主动节能方法	采用预测手段	通过事先预测，节能效果更加明显	需要历史数据进行统计分析，使用场景受限	文献[31,70-71]
	被动节能方法	采用监控手段	容易实现	节能效果不一定最好	文献[26-27,45]
按算法精度分	控制论方法	采用控制理论基础	理论上保证控制精度和系统稳定性	复杂度高，执行时间长	文献[72-76]
	启发式方法	根据直观或经验构造的算法	易于实现，适合于大规模实际环境	不能保证最好的节能效果	文献[19-21,26,30,57-59]
按算法粒度分	细粒度方法	关注单机或系统部件层次	对能耗的局部特征考虑很细	缺乏对全局能耗的考虑	文献[77-79]
	粗粒度方法	关注分布式系统层次	对能耗的全局特征考虑很全	缺乏对局部能耗的考虑	文献[53]
	粗细粒度混合方法	即包含局部，又包含全局层次	综合了细粒度和粗粒度方法的优点	实现相对复杂	文献[19]
按资源种类分	计算资源节能	针对计算资源进行节能	考虑到了计算资源的能耗	无	文献[19-21,41-42,45,51-53]
	存储资源节能	针对存储资源进行节能	考虑到了存储资源的能耗	无	文献[49,54]
	网络资源节能	针对网络资源进行节能	考虑到了网络资源的能耗	无	文献[50]
按应用场景分	节能部署算法	以节能为目标进行虚拟机的部署	考虑到了部署对能耗的影响	无	文献[35,41,53,80-81]
	节能整合算法	以节能为目标进行虚拟机的整合	考虑到了整合对能耗的影响	无	文献[5,52,68,82]
	节能迁移算法	以节能为目标进行虚拟机的迁移	考虑到了迁移对能耗的影响	无	文献[47,83]

YE Ke-Jiang, born in 1986, Ph. D. candidate. His research interests include virtualization and cloud computing, performance evaluation and modeling.

WU Zhao-Hui, born in 1966, Ph. D., professor. His research interests include service science and grid computing, embedded systems, ubiquitous computing, etc.

JIANG Xiao-Hong, born in 1966, Ph. D., associate professor. Her research interests include computer architecture, distributed systems, cloud computing, etc.

HE Qin-Ming, born in 1965, Ph. D., professor. His research interests include virtualization, machine learning, etc.

Background

Power management in the data center is a prominent challenge. With the rapid development of virtualization technology and the emergence of cloud computing paradigm, a large number of future-generation data center will use virtualization technology and cloud computing technology, due to the benefits of high resource utilization, flexible management, and dynamic scalability. Although virtualization holds these benefits, there also exist some challenges, such as power measurement in virtual machines, accuracy of power model, and the trade-off between performance and power, etc.

This paper surveys the latest research results of power management in virtualized cloud computing data centers from the perspectives of power measurement, power modeling, energy efficient mechanisms, and energy efficient algorithms. We firstly analyzed the challenges of operation and power management in virtualized cloud computing platform, then analyzed the difficulties of power monitoring and measurement in virtual machine, after that we introduced the steps of power measurement and power profiling; built the power model for virtual machine system, server consolidation and live migration; summarized the latest advances on power management mechanism from both virtualization level and cloud level; classified and compared the power management algorithms. Finally, we summarized the contents of this paper and proposed ten possible research directions in the future.

This work is supported by National High Technology Research and Development Program (863 Program) of China (No. 2011AA01A207), National Natural Science Foundation of China (No. 11071215), MOE-Intel Information Technology Foundation (No. MOE-INTEL-11-06). These projects aim to study the basic theory and methods of virtualization technology and cloud computing technology. Our group has working in the area of virtualization and cloud computing for several years and published many papers. The papers most related to this work, which focus on the power management in virtualized cloud data center, have been published at Cluster2011, CLOUD2011, GreenCom2010, etc. This paper surveys the power management issues of virtualized cloud computing platform.