

云计算环境中绿色服务级目标的分析、 量化、建模及评价

孙大为^{1,2)} 常桂然³⁾ 陈 东^{1,4)} 王兴伟¹⁾

¹⁾(东北大学信息科学与工程学院 沈阳 110819)

²⁾(清华大学计算机科学与技术系 北京 100084)

³⁾(东北大学计算中心 沈阳 110819)

⁴⁾(麻省大学计算机科学系 阿默斯特 美国 01002)

摘 要 高能效问题是开启绿云计算新纪元的主要障碍之一. 特别是随着全球气候变化和对能量依赖程度的不断增加, 绿色计算作为一种新的高端计算, 受到了人们的广泛关注. 为了实现云计算能耗的最小化, 实现对绿色服务级目标的可量化和可管理, 我们开展了关于云计算环境中绿色服务级目标的分析、量化、建模及评价的研究. 在本文中, 首先给出了绿云的定义. 随后, 通过借鉴服务级目标和绿色计算的相关理论, 对实现绿色服务级目标的原理进行了系统的分析. 之后, 基于绿云的原理和方法, 提出了一种多维能耗模型 M2EC. 理论和实验结果均表明 M2EC 模型在高能效方面具有明显优势, 实现了云计算环境中能耗和服务级目标之间的均衡.

关键词 服务级目标; 高能效; 绿色云; 绿色计算; 云计算; 大数据

中图法分类号 TP393 **DOI 号** 10.3724/SP.J.1016.2013.01509

Profiling, Quantifying, Modeling and Evaluating Green Service Level Objectives in Cloud Computing Environments

SUN Da-Wei^{1,2)} CHANG Gui-Ran³⁾ CHEN Dong^{1,4)} WANG Xing-Wei¹⁾

¹⁾(School of Information Science and Engineering, Northeastern University, Shenyang 110819)

²⁾(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

³⁾(Computing Center, Northeastern University, Shenyang 110819)

⁴⁾(School of Computer Science, University of Massachusetts, Amherst, MA 01002, United States)

Abstract High energy efficiency issue is one of the major obstacles for opening up the new era of the long dreamed vision of computing as a green computing with large-scale cloud data centers. In particular, with the global climate change and the ever growing dependence on energy, green computing, as a high-end computing platform, has received extensive attention. To minimize energy consumption and to achieve the quantifiability and manageability of the high green service level objectives for cloud computing environments, the analysis, quantification, modeling and evaluation of the green service level objectives for cloud computing environments are investigated. In this paper, the definition of green cloud is given and the principles for implementing high green service level objectives are systematically analyzed by referring to the service level objective and green computing theories. Based on the principles and methodology of green cloud, a multi-metric energy consumption model M2EC is put forward. Theoretical as well as experimental results con-

收稿日期: 2011-12-20; 最终修改稿收到日期: 2013-04-24. 本课题得到国家杰出青年科学基金(61225012)、国家自然科学基金(61070162, 71071028, 70931001)、高等学校博士学科点专项科研基金优先发展领域资助课题(20120042130003)、高等学校博士学科点专项科研基金(20100042110025, 20110042110024)、工信部物联网发展专项资金及中央高校基本科研业务费专项资金(N110204003, N120104001)资助.
孙大为, 男, 1985 年生, 博士, 主要研究方向为云计算、大数据、可信计算、绿色计算和信息安全. 常桂然, 男, 1946 年生, 博士, 教授, 博士生导师, 主要研究领域为高性能计算、计算机网络、信息安全. 陈 东, 男, 1983 年生, 博士研究生, 主要研究方向为物联网、云计算. 王兴伟(通信作者), 男, 1968 年生, 博士, 教授, 博士生导师, 主要研究领域为下一代互联网、移动互联网. E-mail: wangxw@mail.neu.edu.cn.

clusively demonstrate that the M2EC algorithm has high potential as it provides efficient green enhancements and significant energy saving. It implements the trade-off between energy consumption and green-service level objectives efficiently and effectively in cloud computing environments.

Keywords service level objectives; high energy efficiency; green cloud; green computing; cloud computing; big data

1 引 言

云计算是以商业目标为主要源动力的大规模分布式计算范式,利用抽象化、虚拟化、透明化、自动化、即时部署、宽带网络等关键技术,通过网络实现资源间的互联、互通和互操作;以统一服务(X as a Service, XaaS)的形式,利用多终端、多平台和多网络,在任意时间、任意地点,并以任意方式为用户提供包括软件、平台和基础设施在内的按需即用、随需应变、快速部署、瞬时释放、物美价廉、安全高效的非平凡 SLOs(Service Level Objectives)服务,其目标是为用户提供高效、灵活且价格低廉的服务;为满足急剧增长的硬件成本、存储能力、服务计算等因互联网的广泛应用所带来的各种新的需求,提供了高效的解决方案,受到政府、工业企业、科研机构、高等院校等的广泛关注^[1-9].

美国国家标准与技术研究院(National Institute of Standards and Technology, NIST)给出了迄今为止最为全面的云计算定义^[5]:“云计算是一种按需即用、随需应变、快速部署、瞬时释放、物美价廉的服务范式,通过网络可以实现对共享计算资源池(如计算、存储、网络、应用、服务)的可访问性、可定制性、可更改性、可付费性和可释放性.同时,在仅需最少的付费、最少的管理、最少的沟通的情况下,即可实现对服务的快速获取、资源的瞬时释放和信息的触手可及”.云计算具有 3 种交互模式、4 种部署模式、5 种公共特征^[1-9]. 3 种交互模式为基础设施即服务、平台即服务和软件即服务;4 种部署模式为公有云、私有云、社区云和混合云;5 种公共特征为按需付费、泛在接入、快速部署、资源透明和标准服务.

在当前云计算的研究过程中,高性能、高可靠、高服务级目标受到了足够的重视,但高能效问题则通常被忽视.据不完全统计^[10-13,19],各类数据中心当前年均电能消耗量占全球年均总电能消耗量的 0.5%.按照当前数据中心规模和数量的发展速度,

预计到 2020 年,数据中心年均电能消耗量将占全球年均总电能消耗量的 1%.数据中心之所以消耗掉如此多的电力资源,主要有两个方面的原因:一方面是由于数据中心的规模和数量与日俱增;另一方面是由于数据中心资源的低效组织与无效管理,如设备大量冗余、制冷系统老化、建筑结构不合理等.因此,随着全球气候变化和对能量依赖程度的不断增加,需要实现云计算环境中能量的有效组织、全面监控、合理管理和高效使用,同时,实现云计算系统高性能和高服务级目标的统一.

“绿色”已经被广泛用于描述事物具有环境友好性特征,如低碳、节能(高能效)、健康、经济、可持续等.绿色计算,即从设计、制造、运输、使用、维护、回收和再利用的整个生命周期过程中,以一种经济、高能效方式使用计算资源.其主要目的是维持三元底线(Triple Bottom Line, 3BL),即人群、地球和利润(People, Planet and Profit, 3P)间的利益均衡^[13-14].绿色云计算,即绿云计算,就是实现云计算的可持续性发展,减少云系统对环境的可能影响.本质而言,云计算本身就是一种高效地实现绿色计算的方式,例如通过虚拟化技术,实现了对资源的有效整合,提高了资源的利用率;通过休眠技术,实现了对能量的有效节省.绿云计算的目标不仅仅要实现计算资源使用效率的极大提高,同时也要实现能量消耗的最小化^[14-19].

为了实现绿云计算,实现云计算的可持续性发展,需要确定云计算环境中整体能量消耗情况.更为重要的是,需要理解如何实现数据中心的能量消耗最小化,高效地实现能量和性能间的均衡,而这些问题恰恰是当前关于云计算能效的研究所缺失的^[14-19].本文系统地分析、量化、建模和评价了大规模数据中心的能量消耗,为实现能量消耗的最小化,实现对高能效的量化、管理打下了基础,同时也满足了用户服务级目标.

本文工作的灵感来自于以下事实:

(1) 资源整合

资源整合是实现数据中心能量节省的一个主要方法,通过对数据中心进行资源整合,可以实现多个不同应用程序共享一台高性能服务器.一方面,可以有效地提高服务器资源的利用率,节省硬件资源的成本,提高资源的使用效率,节省电力成本,并保持系统的负载均衡;另一方面,资源整合也可以为各个应用程序提供一个独立的运行环境,保持应用程序间的独立性和稳定性.

随着硬件技术的不断发展,在当前大多数服务器运行模式中,为了保持各个应用的独立性和稳定性,一台服务器中往往仅仅运行一个应用程序.这样就导致了服务器长期处于较低的使用率,并且,随着服务器硬件性能的改善,服务器的使用效率反而更低.当前情况下,服务器全负荷工作时间仅占全部时间的 10%~30%,台式机的平均使用效率往往低于 5%^[14,15,17-19].然而,通过虚拟化技术,利用虚拟机实现服务器资源的共享,实现服务器资源的整合,服务器的平均使用效率可以提高到 80%以上,能极大地降低系统中能量的消耗.

(2) 在线迁移

在线迁移虚拟机是指在不中断应用服务的情况下,将一个应用及运行该应用的虚拟机从当前运行服务器迁移到另一台服务器中,可以实现全局的负载均衡.或者在可预见的未来,在当前服务器要崩溃的情况下,将当前服务器中运行的应用及运行该应用的虚拟机迁移到其他服务器上,从而满足用户的服务级目标.

(3) 动态资源管理

高能效的动态资源管理也是实现绿云计算的关键技术之一.如图 1(a)所示,在静态的资源管理情况下,当 App2 需要申请新的资源时,即使 App1 已经释放了部分资源,但系统无法将该部分资源动态地分配给 App2 使用.因此,App2 需要向系统申请其他可用资源,或等待 App1 运行结束后,才使用该部分释放的资源.然而,当同样的情况发生在动态资源管理环境中时,如图 1(b)所示,App2 可以动态地使用 App1 随时释放的资源,而不需要向系统申请新的资源,更不需要等到 App1 运行结束后才使用该部分资源.可见动态的资源管理,一方面提高了系统中资源的利用率;另一方面也改善了用户的服务级目标.

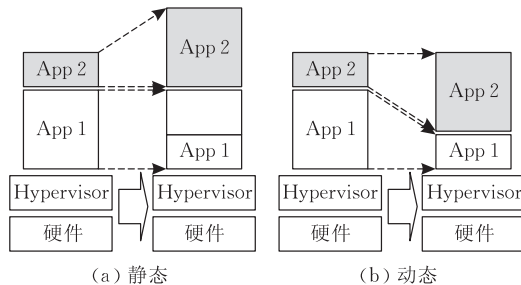


图 1 动态资源管理

(4) 休眠机制

在不影响到服务级目标的前提下,尽可能地将服务器从高能耗状态转换到低能耗状态,以达到降低系统能耗的目标.如图 2 所示,在该 4 种能耗状态下,从唤醒状态到关机状态的转换过程中,能量消耗依次减小,其中,在关机状态下,实现了单位时间内能量消耗的最小化.因此,在任何可能的情况下,在不影响到系统性能的前提下,可以将系统转换到该低能耗状态下,以达到降低系统能耗的目的.

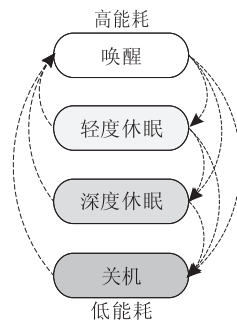


图 2 休眠机制

本文首先给出了绿云的定义,随后通过借鉴服务级目标和绿色计算的相关理论,对实现绿色服务级目标的原理进行了系统的分析.之后提出了一种多维能耗模型 M2EC.本文的主要贡献包括:(1)从量化角度系统地定义了绿云计算、绿云体系结构和绿色服务级目标;(2)分析了云数据中心能量消耗和服务级目标之间的数学关系,得出了满足绿色服务级目标的约束条件;(3)量化了度量绿云计算的多维指标,如能耗、碳排放、健康、经济、可持续性指标,并得出了绿云度;(4)建模了基于 FIFO 和 M2EC 的虚拟机启发式调动算法,量化了虚拟机迁移成本,有效地管理了虚拟机的迁移,通过最小化绿云度,实现了云计算环境中的绿色服务级目标;(5)评价了在大规模云数据中心中基于 M2EC 虚拟机调动的绿色服务级目标,实现了云计算环境中能耗和服务级目标之间的均衡.

本文对云计算环境中的高效性问题进行深入研究. 第 2 节系统建模云计算环境中的高效性, 主要包括绿云计算的基本概念、绿云体系结构和绿色服务级目标; 第 3 节量化多维性能指标和绿云度; 第 4 节具体描述实现高效性的主要策略, 主要包括在线迁移成本量化、虚拟机迁移管理、FIFO 和 M2EC 虚拟机调度算法的描述; 第 5 节详细介绍实验环境、参数设置、实验结果和性能分析. 实验分析表明 M2EC 虚拟机调度算法在绿云度优化、虚拟机负载均衡离差、虚拟机在线迁移率、任务成功执行率等方面具有明显优势; 最后, 第 6 节对本文进行归纳和总结.

2 问题建模

绿云计算, 即实现云计算的可持续性发展, 减少云系统对环境的可能影响, 实现开启未来计算新纪元的目标. 其目的是不仅仅要实现计算资源使用效率的极大提高, 同时也要实现对能量消耗的最小化^[14, 19-27]. 在本节中, 从量化角度定义了绿云计算、绿云度、绿云服务等概念, 详细地分析了绿云体系结构及绿云评价周期, 并明确了云计算环境中的绿色服务级目标.

2.1 绿云计算

绿云计算是一种环境友好型的云计算模式, 为了进行定量的绿色分析, 一个可量化的绿云计算的定义如定义 1 所示.

定义 1. 绿云计算 (green cloud computing). 绿云计算是一个客观评价者 (objective evaluator), 如政府机构、绿色非营利组织、独立的第三方机构等, 关于一个特定的云计算环境的可管理性、可评测性、可维护性等指标进行的层次性的客观评价, 其主要目的是维持三元底线 (即人群、地球和利润) 间的利益均衡. 也就是在设计、制造、运输、使用、维护、回收和再利用的整个生命周期中, 通过多维性能指标, 如低碳、高效、健康、经济、可持续等, 实现对云计算的量化. 一般而言, 绿云计算的绿色程度是通过绿云度 (green cloud degree) 来反映的, 并描述为绿云服务 (green cloud service).

很明显, 在定义 1 中, 将绿云计算视为一个客观的、可量化的、可管理的度量, 其中, “可管理性”、“可评测性”、“可维护性”、“客观评价”等关键词均是重点强调绿云计算的客观性、可量化性、可管理性等特

征. 其次, 绿云计算的目标是实现三元底线的全面负责, 其中, “三元底线”重点强调绿云计算考虑到了人群、地球和利润, 高效地均衡了高效性和绿色服务级目标之间的关系. 该理念最早是在社会科学中描述事物具有很好的环境友好性和可持续性. 再次, 绿云计算可以通过多维性能指标实现度量, 其中, “多维性能指标”是强调绿云计算的度量指标往往不是某个性能指标, 而是需要多个不同方面、不同角度、不同层次的性能指标的协调度量, 才能真实、全面、客观地反映出绿云计算的特征. 通常情况下, 选择低碳、高效、健康、经济、可持续等性能指标实现对绿云计算的度量. 最后, 绿云计算的评价考虑云数据中心生命周期的全部过程, 其中, “整个生命周期过程”是强调绿云计算的评价需要考虑设计、制造、运输、使用、维护、回收和再利用整个生命周期的全部阶段, 而不能简单地选择数据中心生命周期的某个或某些阶段进行绿云计算的评价.

定义 2. 绿云度 (green cloud degree). 绿云度是对云计算环境中绿色程度的一种量化表示, 绿云度 Gd 取值于描述云计算环境绿云度的集合 D , $D \in [0, 1]$. 绿云度 Gd 可以定义为映射 $Gd: f(\mathbf{X}) \mapsto D$, 其中, f 是映射函数, \mathbf{X} 是绿云计算的多维性能指标向量. 具体而言, 绿云度 Gd 可以描述如式 (1) 所示.

$$\begin{cases} Gd = f(\mathbf{X}) = \mathbf{W} \circ \mathbf{X}, \\ \mathbf{W} = (\omega_0, \omega_1, \dots, \omega_{n-1}), \sum_{i=0}^{n-1} \omega_i = 1, \omega_i \in [0, 1], \\ \mathbf{X} = (x_0, x_1, \dots, x_{n-1}) \end{cases} \quad (1)$$

其中, $\mathbf{W} = (\omega_0, \omega_1, \dots, \omega_{n-1})$ 是多维性能指标的权重向量, 当 $\forall \omega_i \in \mathbf{W}$ 时, ω_i 的取值范围在区间 $[0, 1]$, 且满足 $\sum_{i=0}^{n-1} \omega_i = 1$. $\mathbf{X} = (x_0, x_1, \dots, x_{n-1})$ 是多维性能指标向量, 不失一般性, 当 $\forall x_i \in \mathbf{X}$ 时, 设定 x_i 是一个连续量, 且取值范围在区间 $[0, 1]$. 因此, 绿云度 Gd 也将是一个连续量, 且取值范围为 $[0, 1]$.

在一个有五维性能指标的云计算环境中, 当多维性能指标向量 $\mathbf{X} = (0.97, 0.86, 0.99, 0.92, 0.52)$ 时, 并且权重向量 $\mathbf{W} = (0.22, 0.26, 0.31, 0.12, 0.09)$, 那么该云计算环境中的绿云度 $Gd = f(\mathbf{X}) = \mathbf{X} \circ \mathbf{W}^T = 0.9011$.

为了简化问题, 在本文的研究过程中, 仅仅选择云数据中心的 5 个基本的性能指标来实现对绿云计算的评测, 该 5 个性能指标分别为系统能耗 E_{sys} 、系

统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续性 Su_{sys} 。

定义 3. 绿云服务 (green cloud service). 绿云服务 G_s 是对绿云提供商所提供的服务等级的描述, 该服务等级是由绿云度 Gd 所决定的. 绿云服务 G_s 可以描述为一个模糊集, 当 L 是一个包含 n 个绿云服务级的有序评价向量时, 即 $L = (l_0, l_1, l_{n-1})$, 并且满足 $0 \leq l_0 < l_1 < \dots < l_{n-1} \leq 1$. $\forall l_i \in L, l_i$ 的数值越大, 相应的绿云服务等级水平则越高, 那么绿云服务判定函数 Γ 实现了绿云度 Gd 到绿云服务 G_s 的映射, 如式 (2) 所示.

$$G_s = \Gamma(Gd) = \begin{cases} s_0, & 0 \leq Gd < l_0 \\ s_1, & l_0 \leq Gd < l_1 \\ s_{n-1}, & l_{n-2} \leq Gd < l_{n-1} \\ s_n, & l_{n-1} \leq Gd < 1 \end{cases} \quad (2)$$

其中, s_0, s_1, \dots, s_n 是 $n+1$ 个有序绿云服务集, 该绿云服务集的数目 n 可以根据具体的云计算环境需要进行确定.

例如: 一个绿色非营利组织可定义绿云服务判定函数 Γ 如下,

$$G_s = \Gamma(Gd) = \begin{cases} s_0, & 0.00 \leq Gd < 0.32 \\ s_1, & 0.32 \leq Gd < 0.85 \\ s_2, & 0.85 \leq Gd < 1.00 \end{cases}$$

其中, 绿云服务集 $\{s_0, s_1, s_2\} = \{\text{完全非绿色服务, 绿色服务, 完全绿色服务}\}$, 而完全非绿色服务、绿色服务、完全绿色服务的具体规范可以根据具体环境进行确定.

在一个云计算环境中, 当其绿云度 $Gd = 0.15$ 时, $\Gamma(Gd) = \Gamma(0.15) = s_0$, 则意味着该云计算环境将会为其提供完全非绿色服务; 当其绿云度 $Gd = 0.55$ 时, $\Gamma(Gd) = \Gamma(0.55) = s_1$, 则意味着该云计算环境将会为其提供绿色服务; 当其绿云度 $Gd = 0.95$ 时, $\Gamma(Gd) = \Gamma(0.95) = s_2$, 则意味着该云计算环境将会为其提供完全绿色服务.

2.2 绿云体系结构

绿云体系结构可以细分为 4 个空间, 即硬件空间、绿色空间、虚拟空间和用户空间, 如图 3 所示.

在硬件空间中, n 个数据中心分布于不同的地理位置, 在每个数据中心中, 有多个性能不同的物理服务器. 通常情况下, 每个服务器上可以运行一个或多个不同或相同配置的虚拟机系统.

在绿色空间中, 在 VMM/hypervisor 的统一配置和协调之下, 6 个绿色模块协调实现相应的绿色

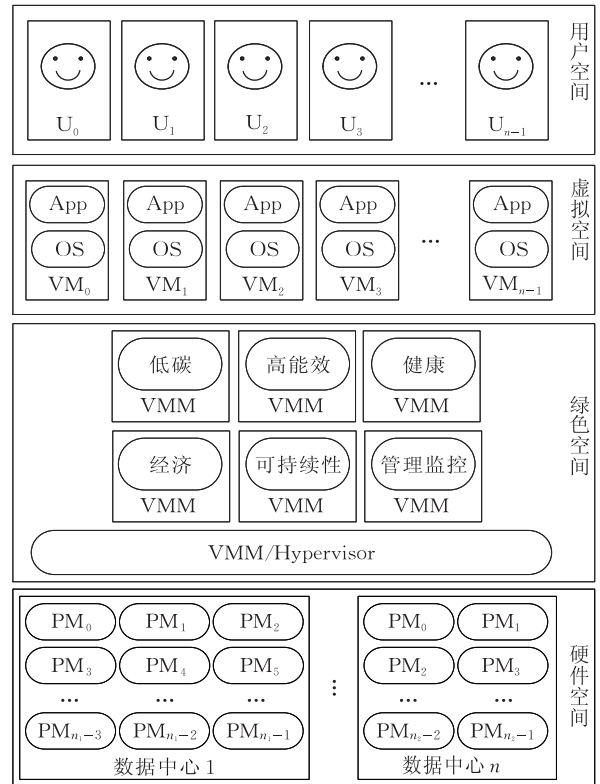


图 3 绿云体系结构

服务级目标. 这 6 个绿色模块分别是低碳模块、高能效模块、健康模块、经济模块、可持续性模块和管理监控模块, 并且分别用于计量碳排放量、能量消耗、辐射等级、投资收益率、可再利用率和进行模块间的管理与监控.

在虚拟空间中, 数据中心可以根据需要创建一个或多个不同配置的虚拟机, 并且可以灵活地实现虚拟机的创建、挂起、迁移或终止等不同能耗状态的转变. 作为用户和数据中心之间的接口, 每台虚拟机可以在一个时间片内接受一个用户的服务请求, 全部虚拟机可以同时并行工作, 并且不同的虚拟机可以根据用户服务请求的需要, 配置不同的操作系统、软件环境和用户软件, 并设置不同的用户服务级目标. 因此, 可以高效地权衡绿色服务和服务级目标之间的关系.

在用户空间中, 一个或多个用户可以在任何时间, 从任何地点, 并以任何方式实现对绿云数据中心的请求和数据访问.

对应于该绿云体系结构的绿云模型如定义 4 所述.

定义 4. 绿云模型 (green cloud model). 绿云模型可以进一步地描述为一个五元组模型, 即 $Gm = (U, DC, G_s, Of, \Theta)$, 其中, $U = \{u_0, u_1, \dots, u_{n-1}\}$ 是

由 n 个用户组成的用户集合,同时,每个用户的服务请求相互独立、互不干扰; $DC = \{dc_0, dc_1, \dots, dc_{n-1}\}$ 是由 c_n 个数据中心组成的数据中心集合,其中,第 i 个数据中心 $dc_i = \{pm_{i_0}, pm_{i_1}, \dots, pm_{i_{n-1}}\}$, pm_{i_k} ($0 \leq k < dc_{n_i}$) 是第 i 个数据中心 dc_i 中的第 k 台物理服务器; G_s 是根据非营利绿色组织所定义的一个绿云服务等级; Of 是为了优化绿云计算环境而定义的目标函数,其定义如式(3)所示,且通过以下 5 个性能指标实现对其进行量化,即系统能耗 E_{sys} 、系统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 、以及系统可持续性 Su_{sys} ; Θ 是实现相应优化策略的优化算法.

$$\begin{cases} Of = Of(E_{sys}, C_{sys}, He_{sys}, Ec_{sys}, Su_{sys}) = \max(Gd_{sys}), \\ \text{s. t. } Gd_{sys} \in [0, 1], \\ \min(E_{sys}), E_{sys} \in [0, 1], \\ \min(C_{sys}), C_{sys} \in [0, 1], \\ \max(He_{sys}), He_{sys} \in [0, 1], \\ \max(Ec_{sys}), Ec_{sys} \in [0, 1], \\ \max(Su_{sys}), Su_{sys} \in [0, 1]. \end{cases} \quad (3)$$

对应于该绿云体系结构的绿云评测周期如定义 5 所述.

定义 5. 绿云评测周期(green cloud evaluate cycle). 一个完整的绿云评测周期可以细分为 4 个阶段,即分析阶段、量化阶段、建模阶段和评价阶段,如图 4 所示. 周期性地通过该 4 个阶段工作,可以实现对绿云计算环境的合理评测,同时可以及时、动态地反馈最新的云计算环境信息.

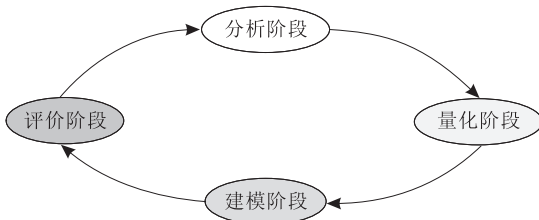


图 4 绿云评测周期

在分析阶段,实现了对数据中心的能量消耗和服务级目标之间的数学关系的分析,得出了满足绿色服务级目标所需要满足的基本条件.

在量化阶段,实现了通过多维性能指标的量化以达到量化绿云计算的功能,同时通过多维性能指标,得出了数据中心的绿云度.

在建模阶段,通过 FIFO 和 M2EC 虚拟机调度算法,实现了对绿云的建模,同时实现了对在线迁移

成本的量化和虚拟化迁移的管理,满足了绿云度最小化的需求,并且也满足了绿色服务级目标的需求.

最后,在评价阶段,实现了对大规模数据中心绿色服务级目标的评价,有效地权衡了绿云计算和服务级目标.

2.3 绿云服务级目标

在用户和服务提供商之间,服务级目标(Service Level Objectives, SLOs)是服务级协议的最核心内容(Service Level Agreement, SLA),且服务级目标可以通过服务级协议的多个不同特征属性实现对其度量,如可用性、可测量性、可控性、可理解性、可支付性等^[28-31].

如图 5 所示,在绿色服务级目标中,不仅仅考虑到了服务级目标,同时也考虑到了绿云计算,实现了对三元底线间的利益均衡,可以通过在云计算环境的全部生命周期过程中的多维性能指标实现对其量化,如碳排放量、高能效、健康性、经济性、可持续性.

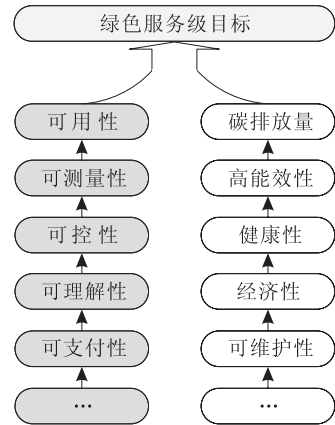


图 5 绿云评测周期

3 绿云量化

本节给出了对云计算环境中 5 个基本性能指标的量化,同时通过组合该 5 个基本性能指标,实现了对云计算环境中绿云度的量化.

3.1 多维性能指标量化

在云计算环境中,多维性能量化指标主要选取以下 5 个基本的性能指标,即系统能耗 E_{sys} 、系统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 和系统可持续性 Su_{sys} .

3.1.1 系统能耗

在云计算环境中,第 i 个数据中心的第 j 个节点 cn_{ij} 的能耗 $E_{cn_{ij}}$,可以通过节点 cn_{ij} 在时间段

$[t_0, t_{n-1}]$ 内的电能消耗量 $P_{cn_{ij}}(\mu(t))$ 计算得出, 如式(4)所示.

$$E_{cn_{ij}} = \int_{t_0}^{t_{n-1}} P_{cn_{ij}}(\mu_{ij}(t)) dt \quad (4)$$

其中, 第 i 个数据中心的第 j 个节点 cn_{ij} 在时间段 $[t_0, t_{n-1}]$ 内的电能消耗量 $P_{cn_{ij}}(\mu(t))$, 可以进一步细分为 $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 和 $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$, 如式(5)所示.

$$P_{cn_{ij}}(\mu_{ij}(t)) = P_{cn_{ij}}^{idl}(\mu_{ij}(t)) + P_{cn_{ij}}^{spd}(\mu_{ij}(t)) \quad (5)$$

其中, $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 是第 i 个数据中心的第 j 个节点 cn_{ij} 的基本(idle state)电能消耗; $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$ 是第 i 个数据中心的第 j 个节点 cn_{ij} 提供服务时所需要具备(spread state)的电量消耗, $\mu_{ij}(t) \in [0, 1]$ 是第 i 个数据中心的第 j 个节点 cn_{ij} 的实时动态变化的负载因子, 在当前数据中心的负载状况下, 负载因子 $\mu_{ij}(t)$ 的实际取值通常在区间 $[0.1, 0.5]$ 内.

节点 cn_{ij} 的电能消耗量 $P_{cn_{ij}}(\mu_{ij}(t))$ 是由 CPU、内存、磁盘等设备的电能消耗量共同决定的^[12, 19, 32, 33]. 由于 CPU 是最主要的电能消耗设备, 为了简化起见, 节点 cn_{ij} 的电能消耗量 $P_{cn_{ij}}(\mu_{ij}(t))$ 可以通过 CPU 的电能消耗量来计算, 如式(6)所示.

$$\begin{cases} P_{cn_{ij}}^{idl}(\mu_{ij}(t)) = \alpha \cdot P_{cn_{ij}}(\max), \alpha \in [0, 1] \\ P_{cn_{ij}}^{spd}(\mu_{ij}(t)) = (1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij}(t), \\ \mu_{ij}(t) \in [0, 1] \end{cases} \quad (6)$$

其中, $P_{cn_{ij}}(\max)$ 是第 i 个数据中心的第 j 个节点 cn_{ij} 在全负荷状态下的最大电能消耗量, $\alpha \in [0, 1]$ 是 $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 和 $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$ 的调节参数, α 的取值由计算机体系结构决定.

$P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 、 $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$ 和 $P_{cn_{ij}}(\mu_{ij}(t))$ 之间的关系^[12, 32-33], 如图 6 所示, 第 i 个数据中心的第 j 个节点 cn_{ij} 的能耗 $E_{cn_{ij}}$ 由定理 1 给出.

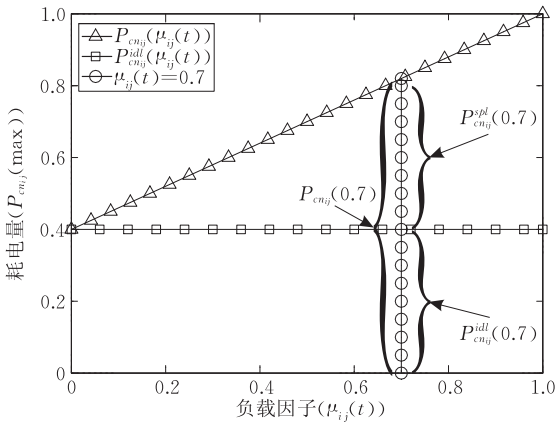


图 6 $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 、 $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$ 和 $P_{cn_{ij}}(\mu_{ij}(t))$ 之间的关系

定理 1. 当时间区间 $[t_0, t_{n-1}]$ 可以进一步细分为 $[t_0, t_1, \dots, t_{n-1}]$ 时, 且 $\forall [t_k, t_{k+1}]$, $k \in [0, 1, \dots, n-1]$, 满足 $\mu_{ij}(t \in [t_k, t_{k+1}]) \equiv \mu_{ij_k}$, 其中, μ_{ij_k} 是不

随时间变化的常量, 且 $\mu_{ij_k} \in [0, 1]$, 那么, 第 i 个数据中心的第 j 个节点 cn_{ij} 的能耗 $E_{cn_{ij}}$ 如式(7)所示.

$$E_{cn_{ij}} = (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + ((1 - \alpha) \cdot P_{cn_{ij}}(\max)) \cdot \sum_{k=1}^{n-1} (\mu_{ij_{k-1}} \cdot (t_k - t_{k-1})) \quad (7)$$

证明. 由于第 i 个数据中心的第 j 个节点 cn_{ij} 的电能消耗量 $P_{cn_{ij}}(\mu_{ij}(t))$ 可以细分为 $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 和 $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$ 两部分, 那么,

$$\begin{aligned} E_{cn_{ij}} &= \int_{t_0}^{t_{n-1}} P_{cn_{ij}}(\mu_{ij}(t)) dt \\ &= \int_{t_0}^{t_{n-1}} (P_{cn_{ij}}^{idl}(\mu_{ij}(t)) + P_{cn_{ij}}^{spd}(\mu_{ij}(t))) dt \\ &= \int_{t_0}^{t_{n-1}} P_{cn_{ij}}^{idl}(\mu_{ij}(t)) dt + \int_{t_0}^{t_{n-1}} P_{cn_{ij}}^{spd}(\mu_{ij}(t)) dt \\ &= \int_{t_0}^{t_{n-1}} \alpha \cdot P_{cn_{ij}}(\max) dt + \int_{t_0}^{t_{n-1}} (1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij}(t) dt \\ &= (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + \int_{t_0}^{t_{n-1}} (1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij}(t) dt \end{aligned}$$

当时间区间 $[t_0, t_{n-1}]$ 可以进一步细分为 $n-1$ 个时间区间 $[t_0, t_1, t_2, \dots, t_{n-2}, t_{n-1}]$ 时, 且 $\forall t \in [t_k, t_{k+1}]$, $i \in [0, 1, \dots, n-1]$, 满足 $\mu_{ij}(t \in [t_k, t_{k+1}]) \equiv \mu_{ij_k}$, 其中, μ_{ij_k} 是不随时间变化的常量, 且 $\mu_{ij_k} \in [0, 1]$, 那么,

$$\begin{aligned} E_{cn_{ij}} &= (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + \int_{t_0}^{t_{n-1}} (1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij}(t) dt \\ &= (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + \int_{t_0}^{t_1} ((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_0}) dt + \int_{t_1}^{t_2} ((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_1}) dt + \dots + \int_{t_{n-2}}^{t_{n-1}} ((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_{n-2}}) dt \\ &= (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + ((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_0}) \cdot (t_1 - t_0) + ((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_1}) \cdot (t_2 - t_1) + \dots + ((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_{n-2}}) \cdot (t_{n-1} - t_{n-2}) \\ &= (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + \sum_{k=1}^{n-1} (((1 - \alpha) \cdot P_{cn_{ij}}(\max) \cdot \mu_{ij_{k-1}}) \cdot (t_k - t_{k-1})) \\ &= (\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + \end{aligned}$$

$$((1-\alpha) \cdot P_{cn_{ij}}(\max)) \cdot \sum_{k=1}^{n-1} (\mu_{ij_{k-1}} \cdot (t_k - t_{k-1})).$$

证毕.

如图 7 所示, 随着参数 α 的不断增大, $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 也将不断增加, 而 $P_{cn_{ij}}^{spd}(\mu_{ij}(t))$ 将不断减少, 但节点 cn_{ij} 的总电能消耗量 $P_{cn_{ij}}(\mu_{ij}(t))$ 最终将不断增加. 同时, 随着负载因子 $\mu_{ij}(t)$ 的不断增大, 节点 cn_{ij} 的总电能消耗量 $P_{cn_{ij}}(\mu_{ij}(t))$ 最终也将不断增加. 可见保持较低的 $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$, 可以有效地减少节点 cn_{ij} 的总电能消耗量 $P_{cn_{ij}}(\mu_{ij}(t))$.

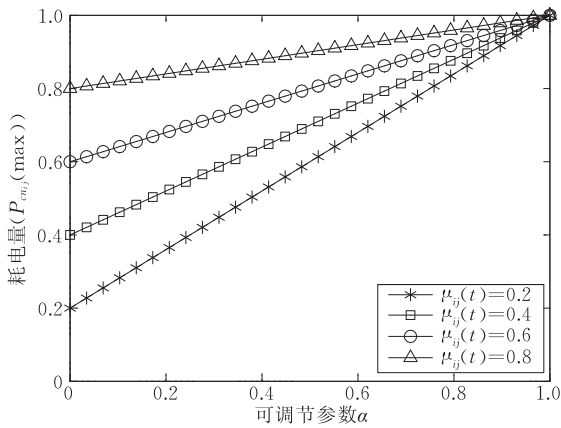


图 7 $P_{cn_{ij}}(\mu_{ij}(t))$ 和 $P_{cn_{ij}}^{idl}(\mu_{ij}(t))$ 之间的关系

定义 6. 系统能耗 (system energy consumption). 系统能耗 E_{sys} 是对云计算环境中每个节点单位时间内的能量消耗的一种度量, 当一个云计算环境由 m 个数据中心组成, 且节点数量向量 $\mathbf{Num} = (num_0, num_1, \dots, num_{m-1})$, 其中 num_i 是第 i 个数据中心所拥有的节点数目, 那么, 整个系统在时间区间 $[t_0, t_{n-1}]$ 内的能耗 E_{sys} 如式(8)所示.

$$E_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} E_{cn_{ij}}}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i} \quad (8)$$

由于第 i 个数据中心的第 j 个节点 cn_{ij} 的最大电能消耗量 $P_{cn_{ij}}(\max)$ 可以根据其体系结构来确定, 那么, 由式(7)可以得出

$$E_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} ((\alpha \cdot P_{cn_{ij}}(\max)) \cdot (t_{n-1} - t_0) + ((1-\alpha) \cdot P_{cn_{ij}}(\max)) \cdot \sum_{k=0}^{n-1} (\mu_{ij_{k-1}} \cdot (t_k - t_{k-1})))}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i}.$$

3.1.2 系统碳排放量

由于第 i 个数据中心的第 j 个节点 cn_{ij} 在时间段 $[t_0, t_{n-1}]$ 内的碳排放量 $C_{cn_{ij}}$ 可以使用相关仪器直接测量出来, 那么, 系统碳排放量 C_{sys} 可由定义 7

给出.

定义 7. 系统碳排放量 (system carbon emission). 系统碳排放量 C_{sys} 是对云计算环境中每个节点单位时间内的碳排放量的一种度量, 当一个云计算环境由 m 个数据中心组成, 且节点数量向量 $\mathbf{Num} = (num_0, num_1, \dots, num_{m-1})$, 其中 num_i 是第 i 个数据中心所拥有的节点数目, 那么, 整个云系统在时间段 $[t_0, t_{n-1}]$ 内的碳排放量 C_{sys} 如式(9)所示.

$$C_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} C_{cn_{ij}}}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i} \quad (9)$$

3.1.3 系统健康

由于第 i 个数据中心的第 j 个节点 cn_{ij} 在时间区间 $[t_0, t_{n-1}]$ 内的实时辐射量 $He_{cn_{ij}}$ 可以使用相关仪器直接测量出来, 那么, 系统健康 He_{sys} 可由定义 8 给出.

定义 8. 系统健康 (system healthful). 系统健康 He_{sys} 是对云计算环境中每个节点单位时间内的辐射量的一种度量, 当一个云计算环境由 m 个数据中心组成, 且节点数量向量 $\mathbf{Num} = (num_0, num_1, \dots, num_{m-1})$, 其中 num_i 是第 i 个数据中心所拥有的节点数目, 那么, 整个系统在时间区间 $[t_0, t_{n-1}]$ 内的系统健康 He_{sys} 如式(10)所示.

$$He_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} He_{cn_{ij}}}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i} \quad (10)$$

3.1.4 系统经济

在云计算环境中, 第 i 个数据中心的第 j 个节点 cn_{ij} 在时间段 $[t_0, t_{n-1}]$ 内的经济效益 $Ec_{cn_{ij}}$ 可由式(11)计算, 那么, 系统经济 Ec_{sys} 由定义 9 给出.

$$Ec_{cn_{ij}} = \frac{R_{cn_{ij}}(t_{n-1}, t_0)}{C_{cn_{ij}}(t_{n-1}, t_0)} \quad (11)$$

其中, $C_{cn_{ij}}(t_{n-1}, t_0)$ 是节点 cn_{ij} 在时间区间 $[t_0, t_{n-1}]$ 内的直接总成本, $R_{cn_{ij}}(t_{n-1}, t_0)$ 是节点 cn_{ij} 在时间区间 $[t_0, t_{n-1}]$ 内的直接总收益. 为了简化起见, 假设 $t_0 = t_{\text{start}}, t_{n-1} \rightarrow +\infty$, 那么, $C_{cn_{ij}}(t_{+\infty}, t_{\text{start}})$ 将是节点 cn_{ij} 购买时的总成本, $R_{cn_{ij}}(t_{+\infty}, t_{\text{start}})$ 将是节点 cn_{ij} 在运行期间内所能实现的直接总收益.

定义 9. 系统经济 (system economics), 系统经济 Ec_{sys} 是对云计算环境中每个节点单位时间内的经济效益的一种度量, 当一个云计算环境由 m 个数据中心组成, 且节点数量向量 $\mathbf{Num} = (num_0, num_1, \dots, num_{m-1})$, 其中 num_i 是第 i 个数据中心

所拥有的节点数目,那么,整个系统在时间区间 $[t_0, t_{n-1}]$ 内的系统经济 Ec_{sys} 如式(12)所示.

$$Ec_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} Ec_{cn_{ij}}}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i} \quad (12)$$

根据式(11)可以得出

$$Ec_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} (R_{cn_{ij}}(t_{n-1}, t_0) / C_{cn_{ij}}(t_{n-1}, t_0))}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i}.$$

3.1.5 系统可持续性

在云计算环境中,第 i 个数据中心的第 j 个节点 cn_{ij} 在时间区间 $[t_0, t_{n-1}]$ 内的可持续性 $Su_{cn_{ij}}$ 可由式(13)计算,那么,系统可持续性 Su_{sys} 可由定义 10 给出.

$$Su_{cn_{ij}} = \frac{R_{cn_{ij}}(t_{n-1}, t_0)}{C_{cn_{ij}}(t_{n-1}, t_0)} \quad (13)$$

其中, $C_{cn_{ij}}(t_{n-1}, t_0)$ 是节点 cn_{ij} 在时间段 $[t_0, t_{n-1}]$ 内的直接总成本, $R_{cn_{ij}}(t_{n-1}, t_0)$ 是节点 cn_{ij} 在时间区间 $[t_0, t_{n-1}]$ 内的可再利用时能获得的总收益. 为了简化起见,假设 $t_0 = t_{\text{start}}, t_{n-1} \rightarrow +\infty$, 那么, $C_{cn_{ij}}(t_{n-1}, t_0)$ 将是节点 cn_{ij} 购买时的总成本, $R_{cn_{ij}}(t_{n-1}, t_0)$ 将是节点 cn_{ij} 被废弃之后能被再利用时所带来的直接总收益.

定义 10. 系统可持续性(system sustainable). 系统可持续性 Su_{sys} 是对云计算环境中每个节点单位时间内的可循环再利用收益的一种度量, 当一个云计算环境由 m 个数据中心组成, 且节点数量向量 $\mathbf{Num} = (num_0, num_1, \dots, num_{m-1})$, 其中 num_i 是第 i 个数据中心所拥有的节点数目, 那么, 整个系统在时间区间 $[t_0, t_{n-1}]$ 内的可持续性 Su_{sys} 如式(14)所示.

$$Su_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} Su_{cn_{ij}}}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i} \quad (14)$$

根据式(13)可以得出

$$Su_{\text{sys}} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{num_i} (R_{cn_{ij}}(t_{n-1}, t_0) / C_{cn_{ij}}(t_{n-1}, t_0))}{(t_{n-1} - t_0) \cdot \sum_{i=0}^{m-1} num_i}.$$

3.2 绿云度量

云计算环境的绿云度 Gd 可以通过以下 5 个基本性能指标计量得出, 即系统能耗 E_{sys} 、系统碳排放

量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续性 Su_{sys} , 也即五维性能指标向量 $\mathbf{X} = (E_{\text{sys}}^{-1}, C_{\text{sys}}^{-1}, He_{\text{sys}}, Ec_{\text{sys}}, Su_{\text{sys}})$. 当每个性能指标均被测量 n 次时, 五维性能指标向量 \mathbf{X} 就转换为矩阵 $\mathbf{M}_{n \times 5} = (p_{ij})_{n \times 5}$. 在更为一般的情况下, 假设有 m 个性能指标, 那么, $\mathbf{M}_{n \times m} = (p_{ij})_{n \times m}$, 其中 p_{ij} 是第 j 个性能指标的第 i 次取值.

一般而言, m 维性能指标之间的度量单位是不统一的, 为了消除不同度量指标之间度量单位的差异性, 需要对矩阵 $\mathbf{M}_{n \times m}$ 进行标准差标准化处理. 另外, 为了保证矩阵 $\mathbf{M}_{n \times m}$ 中所有数据的取值均在区间 $[0, 1]$ 内, 需要对标准差标准化后的矩阵 $\mathbf{M}'_{n \times m}$ 进行极差标准化处理^[34].

对矩阵 $\mathbf{M}_{n \times m}$ 进行标准差标准化处理, 如式(15)所示,

$$p'_{ij} = \frac{p_{ij} - \bar{p}_j}{S_j},$$

$$i \in \{0, 1, \dots, n-1\}, j \in \{0, 1, \dots, m-1\} \quad (15)$$

其中, $\bar{p}_j = \frac{1}{n} \cdot \sum_{i=0}^{n-1} p_{ij}$ 是矩阵 $\mathbf{M}_{n \times m}$ 的第 j 个性能指标的均值, S_j 的定义如式(16)所示, 是矩阵 $\mathbf{M}_{n \times m}$ 的第 j 个性能指标的标准差. 那么, 标准差标准化处理后的矩阵 $\mathbf{M}'_{n \times m} = (p'_{ij})_{n \times m}$.

$$S_j = \sqrt{\frac{1}{n-1} \cdot \sum_{i=0}^{n-1} (p_{ij} - \bar{p}_j)^2}, i \in \{0, 1, \dots, n-1\} \quad (16)$$

经过标准化处理后, 可以保证矩阵 $\mathbf{M}'_{n \times m}$ 中每个性能指标的均值为 0, 方差为 1, 但是并不保证所有数据元素的取值范围在区间 $[0, 1]$ 内. 为了保证矩阵 $\mathbf{M}'_{n \times m}$ 中所有数据的取值均在区间 $[0, 1]$ 内, 需要对标准差标准化后的矩阵 $\mathbf{M}'_{n \times m}$ 通过式(17)进行极差标准化处理, 极差标准化处理后的矩阵 $\mathbf{M}''_{n \times m} = (p''_{ij})_{n \times m}$.

$$p''_{ij} = \frac{p'_{ij} - p'_{\min j}}{p'_{\max j} - p'_{\min j}},$$

$$i \in \{0, 1, \dots, n-1\}, j \in \{0, 1, \dots, m-1\} \quad (17)$$

其中, 参数 $p'_{\min j} = \min(p'_{0j}, p'_{1j}, \dots, p'_{n-1j})$, 参数 $p'_{\max j} = \max(p'_{0j}, p'_{1j}, \dots, p'_{n-1j})$.

m 维性能指标的权重向量 $\mathbf{W} = (\omega_0, \omega_1, \dots, \omega_{m-1})$, 其中 $\omega_i \in [0, 1]$ 是第 j 维性能指标的权重, 且 $\sum_{i=0}^{m-1} \omega_i = 1$.

那么, 最终的绿云度可以通过矩阵 $\mathbf{M}''_{n \times m}$ 和权重向量 \mathbf{W} 相互结合计算得出, 即 $\mathbf{GD} = (Gd_0,$

Gd_1, \dots, Gd_{n-1}) 可以通过式(18)计算得出, 且 $\forall Gd_i \in \mathbf{GD}, Gd_i$ 是在区间 $[0, 1]$ 内的连续量, 系统最终的绿云度是这 n 次处理结果的均值.

$$\mathbf{GD}^T = \mathbf{M}''_{n \times m} \circ \mathbf{W}^T \quad (18)$$

即

$$Gd_i = \sum_{j=0}^{m-1} (p''_{ij} \cdot w_j), j \in \{0, 1, \dots, m-1\} \quad (19)$$

其中, Gd_i 是系统中第 i 次的绿云度处理结果.

4 绿云优化

在本节中, 给出了对在线迁移成本的量化和虚拟机迁移的有效管理方法, 同时通过 FIFO 虚拟机调度算法和 M2EC 启发式虚拟机调度算法实现对绿云度的最小化, 并满足了对绿云计算环境中的绿色服务级目标.

4.1 在线迁移成本量化

在线迁移虚拟机, 是指在不中断应用服务的前提下, 实现将虚拟机从一台服务器迁移到另一台服务器, 满足全局的负载均衡; 或者是在可预见的未来当前服务器可能要崩溃时, 需要将当前服务器上的在线应用迁移到其他可正常提供服务的服务器上来, 以满足用户的服务级目标^[35-45].

然而在在线迁移的过程中, 也将会影响到当前迁移虚拟机中所正在运行的应用服务. 一方面, 由于在线迁移过程包括以下 6 个步骤: 预迁移 (pre-migration)、等待迁移 (reservation)、预拷贝 (iterative pre-copy)、拷贝 (stop-and-copy)、迁移 (commitment) 和激活 (activation), 而这 6 步均需要消耗掉一定的时间, 因此, 势必将增加待迁移虚拟机中的在线服务的完成时间. 另一方面, 由于待迁移虚拟机中的服务可能因为缺少所需要的资源而被迫中止, 因此在线迁移过程中也可能会影响到在线服务的服务质量. 可见需要实现对在线迁移的时间成本和性能成本进行量化, 实现相关成本的最小化, 才能改善云计算环境中的绿云度 Gd .

虚拟机 VM_i 在线迁移的时间成本 tc_{VM_i} 可以细分为 3 部分, 如式(20)所示, 即关机 (shut down) 时间成本 $tc_{VM_i}^{sd}$ 、迁移持续 (migration duration) 时间成本 $tc_{VM_i}^{mg}$ 以及启动 (start up) 时间成本 $tc_{VM_i}^{su}$. 同时, 该时间成本与运行于虚拟机之上的应用服务无关, 而只与虚拟机本身的特征有关.

$$tc_{VM_i} = tc_{VM_i}^{sd} + tc_{VM_i}^{mg} + tc_{VM_i}^{su} \quad (20)$$

其中, 关机时间成本 $tc_{VM_i}^{sd}$ 和启动时间成本 $tc_{VM_i}^{su}$ 对于一种特定的虚拟机而言将是一个固定值, 即为常量;

迁移持续时间成本 $tc_{VM_i}^{mg}$ 与虚拟机 VM_i 所需要迁移的文件大小 S_{VM_i} 以及当前系统可用带宽 B_{VM_i} 之间的关系如式(21)所示.

$$\begin{cases} tc_{VM_i}^{sd} = \Delta_1 tc_{VM_i}, \\ tc_{VM_i}^{mg} = \frac{S_{VM_i}}{B_{VM_i}}, \\ tc_{VM_i}^{su} = \Delta_2 tc_{VM_i}. \end{cases} \quad (21)$$

虚拟机 VM_i 在线迁移的性能成本 pc_{VM_i} 如式(22)所示^[36-39].

$$pc_{VM_i} = \theta_i \cdot \int_{t_0}^{t_{n_i}} u_{VM_i} dt \quad (22)$$

其中, θ_i 是调节参数, 通常取值 $0, 1$, t_0 是虚拟机 VM_i 开始迁移的时间, t_{n_i} 是虚拟机 VM_i 结束迁移的时间, 且 $t_n = t_0 - tc_{VM_i}$, u_{VM_i} 是虚拟机 VM_i 的 CPU 利用率.

定理 2. 设在当前时间片内共有 m 台虚拟机需要被在线迁移, 并且将所有的调节参数 $\theta_0, \theta_1, \dots, \theta_m$ 均设置为 θ . 那么, 系统全部的性能成本 Pc_{VMs} 如式(23)所示.

$$Pc_{VMs} = \theta \cdot \sum_{i=0}^m \left(\int_{t_0}^{t_{n_i}} u_{VM_i} dt \right) \quad (23)$$

证明. 由于当前时间片内共有 m 台虚拟机需要被在线迁移, 那么,

$$Pc_{VMs} = \sum_{i=0}^m pc_{VM_i}.$$

根据式(22),

$$Pc_{VMs} = \sum_{i=0}^m pc_{VM_i} = \sum_{i=0}^m \left(\theta_i \cdot \int_{t_0}^{t_{n_i}} u_{VM_i} dt \right).$$

当将所有的调节参数均设置为 θ 后, 即

$$\theta_0 = \theta_1 = \dots = \theta_m = \theta,$$

那么,

$$Pc_{VMs} = \sum_{i=0}^m \left(\theta_i \cdot \int_{t_0}^{t_{n_i}} u_{VM_i} dt \right) = \theta \cdot \sum_{i=0}^m \left(\int_{t_0}^{t_{n_i}} u_{VM_i} dt \right).$$

证毕.

4.2 虚拟机迁移管理

迁移虚拟机的目标是为了满足一定的绿色服务级目标的要求, 因此在迁移虚拟机时需要考虑以下两个情况:

(1) 如图 8 所示, 当服务器 pm_i 处于较低的负载状态时, 若数据中心的其他服务器可以接受服务器 pm_i 上的全部 m 台虚拟机, 并且该 m 台虚拟机的迁移成本 Pc_{VMs} 小于服务器 pm_i 运行该 m 台虚拟机的运行成本 Rc_{pm_i} , 如式(24)所示, 那么, 可以考虑迁移全部 m 台虚拟机.

$$\begin{cases} Rc_{pm_i} > \tau \cdot Pc_{VMs}, \text{ 或者,} \\ Rc_{pm_i} > \tau \cdot \left(\theta \cdot \sum_{i=0}^m \left(\int_{t_0}^{t_{n_i}} u_{VM_i} dt \right) \right), \tau \geq 1 \end{cases} \quad (24)$$

其中, τ 是判别是否进行在线迁移的参数, 可以根据需要动态设置. 当 $\tau=1$ 时, 即一旦 m 台虚拟机的迁移成本 $P_{c_{VMs}}$ 小于服务器 pm_i 运行该 m 台虚拟机的运行成本 $R_{c_{pm_i}}$ 时, 就立即迁移出全部 m 台虚拟机.

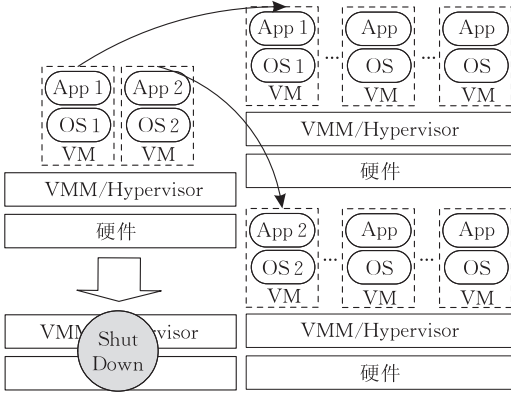


图 8 在线迁移(低负载状态下)

迁移出全部 m 台虚拟机, 并且关闭之前运行的服务器 pm_i , 所能够节省的成本为 $R_{c_{pm_i}} - \tau \cdot P_{c_{VMs}}$, 因此, 可以在一定程度上改善系统的绿云度.

(2) 当服务器 pm_i 处于一个较高的负载状态 Ld_{pm_i} , 且已经不能够满足系统的服务级目标 $SLOs_u$ 的要求, 如式(25)和图 9 所示, 或者服务器 pm_i 在未来一个可以预测的时间内即将崩溃, 在这种情况下也需要迁移出服务器 pm_i 上部分或全部 m 台虚拟机.

$$\begin{cases} Ld_{pm_i} > \overline{Ld}_{sys} + \epsilon \cdot \sigma, \epsilon \geq 1 \text{ 且} \\ SLOs_{pm_i} < SLOs_u \end{cases} \quad (25)$$

其中, ϵ 是判别是否进行在线迁移的参数, 可以根据实际系统的需要进行设置, \overline{Ld}_{sys} 是整个数据中心各台服务器负载均值, σ 是整个数据中心各台服务器负载标准差, $SLOs_{pm_i}$ 是服务器 pm_i 在当前负载状态 Ld_{pm_i} 下所可能提供的服务级目标.

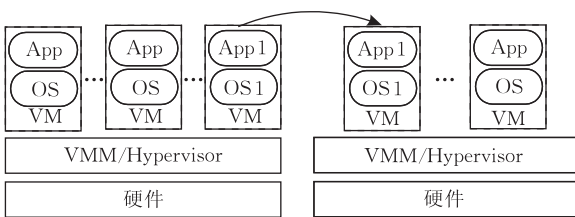


图 9 在线迁移(高负载状态下)

通过在线迁移服务器 pm_i 上的部分或全部 m 台虚拟机, 可以显著地改善服务器 pm_i 上的服务级目标. 然而, 需要迁移的虚拟机的具体数目, 则需要权衡绿云度和服务级目标之间的关系. 在满足一定的服务级目标的情况下, 尽可能少地迁移虚拟机将是一个较优的选择, 因为每多迁移一台虚拟机, 势必将多增加一个单位的性能成本 $p_{c_{VMs}}$.

4.3 FIFO 虚拟机调度算法

先来先服务虚拟机调度算法 (Fisrt In, First Out, FIFO) 是一种简单的根据服务请求的到达时间来为该服务分配虚拟机的算法, 也是一种最原始、最简单、最基本的虚拟机调度算法^[36-39].

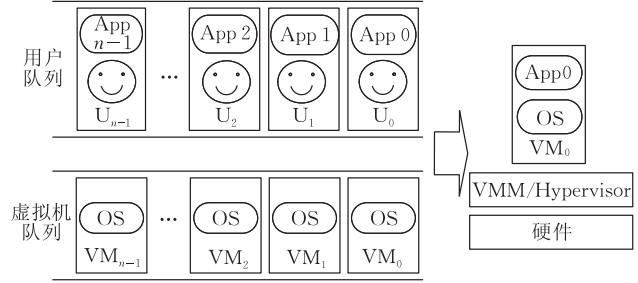


图 10 FIFO 虚拟机调度算法

如图 10 所示, FIFO 虚拟机调度算法只是简单地从用户请求队列 Q_{Userss} 的最早到达的服务请求, 即队头中, 取出一个应用 App_0 , 然后将 App_0 运行于虚拟机队列 Q_{Vms} 的队头虚拟机 VM_0 中. 由于 FIFO 虚拟机调度算法没有考虑到任何的优化机制, 对应用和虚拟机的选择, 仅仅是根据应用和虚拟机到达队列的时间来确定的. 因此, 使用 FIFO 虚拟机调度算法时的系统绿云度 Gd_{FIFO} 可以作为系统的基本绿云度, FIFO 虚拟机调度算法可以描述为如算法 1 所示.

算法 1. FIFO 调度算法.

输入: user queue Q_{Userss} , virtual machine queue Q_{Vms} , physical machine queue Q_{Pms} , five-metric performance vector $\mathbf{X} = (E_{sys}^{-1}, C_{sys}^{-1}, He_{sys}, Ec_{sys}, Su_{sys})$, and weight vector $\mathbf{W} = (\omega_E, \omega_C, \omega_{He}, \omega_{Ec}, \omega_{Su})$

输出: green cloud degree Gd_{FIFO}

1. If user queue Q_{Userss} is empty at current time slice ts_0 then
2. Wait a time slice.
3. End if
4. While user queue Q_{Userss} is not empty at current time slice ts_0 do
5. Fetch the first application App_0 in user queue Q_{Userss} .
6. If virtual machine queue Q_{Vms} is empty then
7. Monitor the virtual machine queue Q_{Vms} until new virtual machine available.
8. End if
9. Fetch the first virtual machine VM_0 in virtual machine queue Q_{Vms} .
10. Run the fetched App_0 on the fetched VM_0 .
11. Allocate the VM_0 to a Physical machine PM of the physical machine queue Q_{Pms} .

12. End while
13. Calculate energy consumption E_{sys} , system carbon emission C_{sys} , system health effect He_{sys} , system economics Ec_{sys} and system sustainability Su_{sys} by (8), (9), (10), (12), and (14), respectively.
14. Calculate green cloud degree Gd_{FIFO} by (18).
15. Return Gd_{FIFO} .

FIFO 虚拟机调度算法的输入分别是用户队列 Q_{Users} 、虚拟机队列 Q_{Vms} 、服务器队列 Q_{Pms} 、五维性能向量 $\mathbf{X}=(E_{sys}^{-1}, C_{sys}^{-1}, He_{sys}, Ec_{sys}, Su_{sys})$ 和权重向量 $\mathbf{W}=(\omega_E, \omega_C, \omega_{He}, \omega_{Ec}, \omega_{Su})$, FIFO 虚拟机调度算法的输出是绿云度 Gd_{FIFO} . 在 FIFO 虚拟机调度算法中, 从步 4 到步 12, 实现了从用户队列 Q_{Users} 中取出相应的队头的用户服务请求, 并分配到虚拟机队列 Q_{Vms} 中的队头的虚拟机上; 步 13 分别计算系统能耗 E_{sys} 、系统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续 Su_{sys} ; 步 14 计算该云计算环境的绿色度 Gd_{FIFO} . FIFO 虚拟机调度算法的时间复杂度为 $O(1)$.

4.4 M2EC 虚拟机调度算法

多维能耗虚拟机调度算法 (Multi-Metric Energy Consumption virtual machines scheduling algorithm, M2EC) 是基于首次适合下降算法 (First Fit Decreasing algorithm, FFD) 的思想, 再次改进和开发出来的. 由于 FFD 算法具有很好的理论基础, 在实践中得到了广泛应用. 但 FFD 算法具有负载不均衡性特征, 因此重点是对该不足进行必要的调整和完善^[27, 35-36, 41-48]. M2ME 虚拟机调度算法如算法 2 所示.

算法 2. M2EC 调度算法.

输入: user queue Q_{Users} , virtual machine queue Q_{Vms} , physical machine queue Q_{Pms} , five-metric performance vector $\mathbf{X}=(E_{sys}^{-1}, C_{sys}^{-1}, He_{sys}, Ec_{sys}, Su_{sys})$, and weight vector $\mathbf{W}=(\omega_E, \omega_C, \omega_{He}, \omega_{Ec}, \omega_{Su})$

输出: green cloud degree Gd_{M2EC}

1. If user queue Q_{Users} is empty at current time slice t_{s_0} then
2. Wait a time slice.
3. End if
4. If virtual machine queue Q_{Vms} is empty at current time slice t_{s_0} then
5. Wait a time slice.
6. End if
7. While user queue Q_{Users} is not empty at current time slice t_{s_0} do
8. Fetch an application App in user queue Q_{Users} and
Fetch a virtual machine VM in virtual machine

queue Q_{Vms} .

9. Run the fetched App on the fetched VM.
10. End while
11. Sort the virtual machines in the queue Q_{Vms} by performance cost $\rho_{c_{VM}}$ in decreasing order.
12. Sort the physical machines in the queue Q_{Pms} by the remaining available resources in decreasing order.
13. For each virtual machine in queue Q_{Vms} do
14. For each physical machine in queue do
15. If the remaining available resources are more than the resources needed by the VM then
16. Allocate the VM to a Physical machine PM.
17. Update the remaining available resources of the allocated Physical machine PM.
18. End if
19. Break.
20. End for
21. End for
22. Calculate the energy consumption E_{sys} , system carbon emission C_{sys} , system health effect He_{sys} , system economics Ec_{sys} and system sustainability Su_{sys} by (8), (9), (10), (12), and (14), respectively.
23. Calculate the green cloud degree Gd_{M2EC} by (18).
24. Return Gd_{M2EC} .

M2EC 虚拟机调度算法的输入分别是用户队列 Q_{Users} 、虚拟机队列 Q_{Vms} 、服务器队列 Q_{Pms} 、五维性能向量 $\mathbf{X}=(E_{sys}^{-1}, C_{sys}^{-1}, He_{sys}, Ec_{sys}, Su_{sys})$ 和权重向量 $\mathbf{W}=(\omega_E, \omega_C, \omega_{He}, \omega_{Ec}, \omega_{Su})$, M2EC 虚拟机调度算法的输出是绿云度 Gd_{M2EC} . 在 M2EC 虚拟机调度算法中, 从步 7 到步 10, 实现了接收用户的服务请求, 并分配到虚拟机上; 步 11 对虚拟机队列 Q_{Vms} 按性能降序排列; 步 12 对服务器队列 Q_{Pms} 按性能降序排列; 从步 13 到步 21, 使用 FFD 算法的基本思想, 为虚拟机选择并分配一个最合适的服务器; 步 22 分别计算系统能耗 E_{sys} 、系统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续 Su_{sys} ; 步 23 计算绿色度 Gd_{M2EC} . M2EC 虚拟机调度算法的时间复杂度为 $O(n \log n)$.

5 实验与性能分析

为了分析所提出的 M2EC 虚拟机调度算法的性能, 本节介绍实验过程中的详细的模拟环境和相关参数的设置, 然后给出了实验性能的对比测试和分析结果. 其中, 测试和分析的性能指标为绿云度 Gd 、虚拟机负载均衡离差 σ 、虚拟机在线迁移率 ldr 和任务成功执行率 ser .

5.1 实验环境与参数设置

我们选择 CloudSim Toolkit^[49-51] 作为实验模拟平台, 该模拟平台基于 Java 开发, 可以有效地进行大规模云计算数据中心的模拟和建模. 其中, 云计算环境中的用户和资源的特殊需求可以通过重写相应的接口得以实现, 该特性可以有效地构建云计算环境中不同用户和资源的需要. 此外, CloudSim Toolkit 是在 GridSim Toolkit 的基础上开发和发展起来的, 而 GridSim Toolkit 可以有效地支持建模和构建多网格接口的高级别软件构建. 而 GridSim Toolkit 是基于 SimJava Toolkit 开发的, SimJava Toolkit 可以为高层次的模拟架构提供分散事件模拟引擎, 并且可以有效地通过多线程模拟不同实体, 该特性可以有效地满足云计算环境中相应实体的随机特性. 因此, 选择 CloudSim Toolkit 作为实验的模拟平台, 可以较好地测试本文提出的 M2EC 虚拟机调度算法的相关性能, 并且可以较好地反映实际云计算环境的相关情况.

在实验过程中, 共创建了 10 个云数据中心, 每个云数据中心的详细参数配置如表 1 所示. 同时, 设置了 500 台虚拟机来提供云服务, 每台虚拟机的 PE (Processing Elements) 数目在区间 [2, 4] 内.

表 1 数据中心的详细配置

数据中心 ID	机器数	每台机器的 PE 数	处理能力 (MIPS)	带宽 / Gbps
DC_0~1	10	2~4	200~400	1.0
DC_2~3	20	2~4	400~600	2.0
DC_4~5	30	4~8	600~800	5.0
DC_6~7	40	8~16	800~1000	5.0
DC_8~9	50	12~16	800~1000	10.0

此外, 在实验过程中, 提交了 1000 个任务到 500 台虚拟机上, 每个任务的提交时间服从泊松分布, 每个任务的计算长度在区间 [10 000, 20 000] MI 内. 实验过程测试了 5 个基本性能指标, 即系统能耗 E_{sys} 、系统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续 Su_{sys} . 实验的详细参数配置如表 2 所示.

表 2 模拟参数配置

参数	取值	描述
α	0.75	公式(6)中的参数
$\omega_0, \omega_1, \omega_2, \omega_3, \omega_4$	分别为 0.22, 0.26, 0.31, 0.12, 0.09	公式(18)中的 5 个权重
θ	0.10	公式(23)中的参数
τ	2.00	公式(24)中的参数
ϵ	3.00	公式(25)中的参数

5.2 实验结果与性能分析

对实验过程中测试和分析的 4 个主要相关指标, 分别说明如下.

(1) 绿云度 Gd , 即云计算系统中绿色程度的一种度量. 绿云度 Gd 可以通过系统能耗 E_{sys} 、系统碳排放量 C_{sys} 、系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续 Su_{sys} 5 个基本性能指标综合计算得出. 一个云系统的绿云度 Gd 越大, 反映该系统能耗 E_{sys} 和系统碳排放量 C_{sys} 越小, 系统健康 He_{sys} 、系统经济 Ec_{sys} 以及系统可持续 Su_{sys} 则越大, 那么, 就可以在更大的程度上满足用户的绿色服务级目标, 如式(18)所示.

如图 11 所示, 随着任务负载比例的不断加, M2EC 虚拟机调度算法可以保证系统的绿云度 Gd 处于较高的水平, 即使是在系统中任务的负载比例高达 90% 以上时, 仍然可以保证系统的绿云度 Gd 高达 0.72. 与 FIFO 虚拟机调度算法相比较, M2EC 虚拟机调度算法可以明显改善整个系统的绿云度 Gd , 极大地改进了整个系统的绿色服务级目标, 能很好地满足用户的需求.

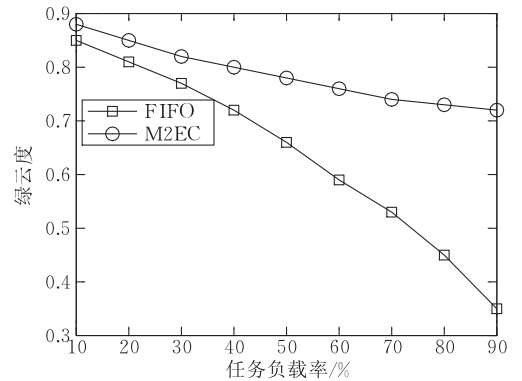


图 11 绿云度

(2) 虚拟机负载均衡离差 σ , 反映了随着用户请求服务的不断变化, 在在线迁移后的整个系统中的虚拟机的负载均衡情况. 虚拟机负载均衡离差 σ 越小, 表明整个系统中虚拟机在各台服务器间的负载越均衡, 越能较好地满足绿色服务级目标, 如式(26)所示.

$$\sigma = \sqrt{\frac{1}{n-1} \cdot \sum_{i=0}^{n-1} (ld_i - \bar{ld})^2} \quad (26)$$

其中, ld_i 如式(27)所示,

$$ld_i = \text{num}(VMs)_i / \text{num}(PEs)_i \quad (27)$$

其中, $\text{num}(VMs)_i$ 是第 i 台服务器所拥有的虚拟机数目, $\text{num}(PEs)_i$ 是第 i 台服务器所拥有的处理单元 PE 数目.

如图 12 所示,随着任务负载比例的不断加, M2EC 虚拟机调度算法可以保证负载均衡离差处于一个相对较低的水平. 即使是在系统中任务的负载比例高达 90% 以上时, 仍然可以保证系统的虚拟机负载均衡离差 σ 位于 0.349 这一相对较低的水平之下. 与 FIFO 虚拟机调度算法相比, M2EC 虚拟机调度算法可以明显改善虚拟机负载均衡离差 σ , 满足了用户的绿色服务级目标.

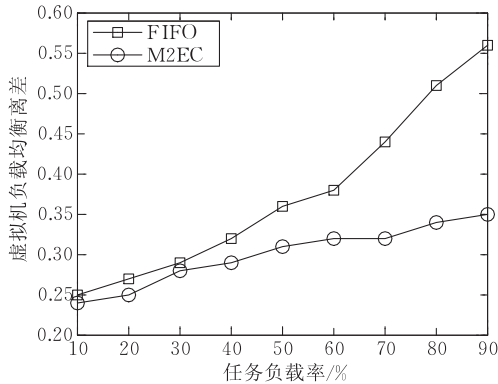


图 12 虚拟机负载均衡离差

(3) 通过在线迁移虚拟机, 实现虚拟机在整个系统中各台服务器间的负载均衡, 将可以极大地改善用户的绿色服务级目标. 虚拟机在线迁移率 ldr 反映了整个系统中虚拟机在线迁移的比值, ldr 越小, 相应的虚拟机迁移成本则越小, 必将进一步改善整个系统的绿色度 Gd . 具体而言, 虚拟机在线迁移率 ldr 如式(28)所示.

$$ldr = \frac{1}{n} \cdot \sum_{i=0}^{n-1} \xi_i, \quad \xi_i = \begin{cases} 1, & \text{如果 } vm_i \text{ 被迁移} \\ 0, & \text{其它} \end{cases} \quad (28)$$

其中, n 是在整个系统服务过程中所创建的虚拟机数目.

如图 13 所示, 随着任务负载比例的不断加, M2EC 虚拟机调度算法可以保证系统的虚拟机在线迁移率 ldr 处于较低的水平. 即使是在系统中任务的负载比例高达 90% 以上时, 仍然可以保证 ldr 位于 0.21 这一相对较低的水平. 与 FIFO 虚拟机调度算法相比, M2EC 虚拟机调度算法在付出较小的成本的情况下, 将极大地改进了整个系统的绿色服务级目标. 特别是在系统的负载比例较高的情况下, 这种优势将更加明显.

(4) 任务的成功执行是指一个任务在所需要的最短结束时间之前被成功完成, 任务成功执行率 ser 是指在最短结束时间之前成功完成的任务和系统中用户请求服务的全部任务之间的比值. 在一个云计

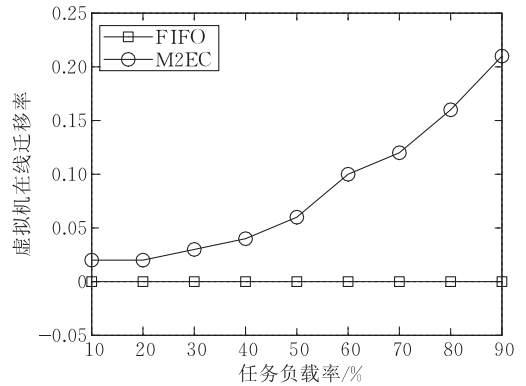


图 13 虚拟机在线迁移率

算环境中, 任务成功执行率 ser 越高, 用户的绿色服务级目标将越容易满足. 具体而言, 任务成功执行率 ser 如式(29)所示.

$$ser = \frac{\sum_{j=0}^{m-1} \sum_{k=0}^{m_j-1} \xi_{j_k}}{\sum_{j=0}^{m-1} m_j}, \quad \xi_{j_k} = \begin{cases} 1, & \text{如果任务 } t_{j_k} \text{ 完成} \\ 0, & \text{其它} \end{cases} \quad (29)$$

其中, m_j 是用户 j 所请求服务的任务总数.

如图 14 所示, 在没有部署 M2EC 虚拟机调度算法的情况下, 随着任务负载比例的不断加, 整个系统的任务成功执行率 ser 急剧下降, 特别是当系统的任务负载比例高于 50% 的情况下, 这种趋势则更加明显. 当系统部署了 M2EC 虚拟机调度算法后, M2EC 虚拟机调度算法可以保证系统的任务成功执行率 ser 处于一个相对较高的、较稳定的水平, 即使是在系统中任务的负载比例高达 90% 以上时, 仍然可以保证系统的任务成功执行率 ser 位于 0.951 这一相对较高的水平. 可见 M2EC 虚拟机调度算法可以在很大程度上改善整个系统的任务成功执行率 ser , 满足用户的绿色服务级目标.

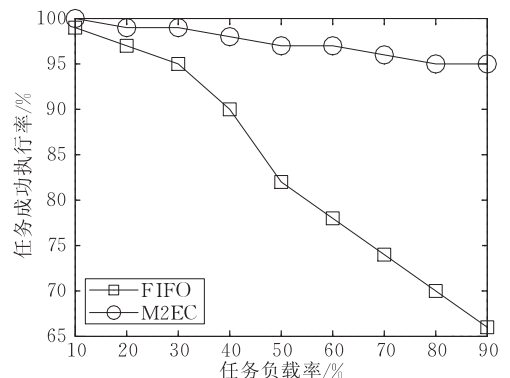


图 14 任务成功执行率

6 总 结

本文对云计算环境中的高能效性问题进行了深入研究, 给出了云计算环境中的高能效性概念模型, 主要包括绿云计算的基本概念、绿云体系结构和绿色服务级目标. 为定量评估绿色服务的等级, 给出了量化的多维性能指标和绿云度. 具体描述了实现高能效性的绿云优化策略, 主要包括在线迁移成本量化、虚拟机迁移管理、FIFO 虚拟机调度算法和 M2EC 虚拟机调度算法. 通过实验分析表明, M2EC 虚拟机调度算法在绿云度优化、虚拟机负载均衡离差、虚拟机在线迁移率、任务成功执行率等方面具有明显优势.

在将来的研究工作中, 我们将进一步深化绿云度量、量化指标、优化策略等内容. 同时, 结合具体的云计算环境, 在大规模云数据中心进行部署和实践.

参 考 文 献

- [1] Armbrust M, Fox A, Griffith R, et al. A view of cloud computing. *Communications of the ACM*, 2010, 53(4): 50-58
- [2] Marston S, Li Z, Bandyopadhyay S, et al. Cloud computing—The business perspective. *Decision Support Systems*, 2011, 51(1): 176-189
- [3] Buyya R, Yeo C S, Venugopal S, et al. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 2009, 25(6): 599-616
- [4] Truong H L, Dustdar S. Cloud computing for small research groups in computational science and engineering: current status and outlook. *Computing*, 2011, 91(1): 75-91
- [5] Mell P, Grance T. The NIST definition of cloud computing. *Communications of the ACM*, 2010, 53(6): 50
- [6] Li Wei, Hu Song-Lin, Liu Dong-Mei, et al. Community clustering of green messaging system in cloud computing. *Chinese Journal of Computers*, 2012, 35(6): 1327-1336 (in Chinese)
(李伟, 虎嵩林, 刘冬梅等. 云计算环境下基于社区聚集的绿色消息系统. *计算机学报*, 2012, 35(6): 1327-1336)
- [7] Cheng Fang-Quan, Peng Zhi-Yong, Song Yong, et al. An efficient privacy-preserving rank query over encrypted data in cloud computing. *Chinese Journal of Computers*, 2012, 35(11): 2215-2227 (in Chinese)
(程芳权, 彭智勇, 宋伟等. 云环境下一种隐私保护的高效密文排序查询方法. *计算机学报*, 2012, 35(11): 2215-2227)
- [8] Wang Shan, Wang Hui-Ju, Qin Xiong-Pai, et al. Architecting big data: Challenges, studies and forecasts. *Chinese Journal of Computers*, 2011, 34(10): 1741-1754 (in Chinese)
(王珊, 王会举, 覃雄派等. 架构大数据: 挑战、现状与展望. *计算机学报*, 2011, 34(10): 1741-1754)
- [9] Gabriel M, Wolfgang G, Calvin J R. Hybrid computing—Where HPC meets grid and cloud computing. *Future Generation Computer Systems*, 2011, 27(5): 440-453
- [10] Lee Y C, Zomaya A Y. Energy conscious scheduling for distributed computing systems under different operating conditions. *IEEE Transactions on Parallel and Distributed Systems*, 2011, 22(8): 1374-1381
- [11] Li J, Li B, Wo T, et al. CyberGuarder: A virtualization security assurance architecture for green cloud computing. *Future Generation Computer Systems*, 2012, 28(2): 379-390
- [12] Von Laszewski G, Wang L, Younge A, et al. Power-aware scheduling of virtual machines in DVFS-enabled clusters// *Proceedings of the 2009 IEEE International Conference on Cluster Computing and Workshops, CLUSTER 2009*. New Orleans, Louisiana, USA, 2009: 144-153
- [13] Shuman D I, Liu M, Wu O Q. Energy-efficient transmission scheduling with strict underflow constraints. *IEEE Transactions on Information Theory*, 2011, 57(3): 1344-1367
- [14] Talebi M, Way T. Methods, metrics and motivation for a Green computer science program. *Sigcse Bulletin*, 2009, 41(1): 362-366
- [15] Baliga J, Ayre R W A, Hinton K, et al. Green cloud computing: Balancing energy in processing, storage, and transport. *Proceedings of the IEEE*, 2011, 99(1): 149-167
- [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 2010 International Conference on Parallel and Distributed Processing Techniques and Applications*. Las Vegas, USA, 2010: 1-12
- [17] Joshi Y, Samadiani E. Energy efficient thermal management of data centers via open multi-scale design: A review of research questions and approaches. *Journal of Enhanced Heat Transfer*, 2011, 18(1): 15-30
- [18] Carter J, Rajamani K. Designing energy-efficient servers and data centers. *Computer*, 2010, 43(7): 76-78
- [19] 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)
- [20] Lefèvre L, Orgerie A C. Designing and evaluating an energy efficient Cloud. *Journal of Supercomputing*, 2010, 51(3): 352-373
- [21] Lee Y C, Zomaya A Y. Energy efficient utilization of resources in cloud computing systems. *Journal of Supercomputing*, 2012, 60(2): 268-280
- [22] Klizaovich D, Bouvry P, Khan S U. GreenCloud: A packet-level simulator of energy-aware cloud computing data centers// *Proceedings of the 2010 IEEE Global Communications*

- Conference, GLOBECOM 2010, Miami, USA, 2010: 1-5
- [23] Lang W, Patel J M. Energy management for MapReduce clusters. *VLDB Endowment*, 2010, 3(1-2): 129-139
- [24] Jang J W, Jeon M, Kim H S, et al. Energy reduction in consolidated servers through memory-aware virtual machine scheduling. *IEEE Transactions on Computers*, 2011, 60(4): 552-564
- [25] Garg S K, Yeo C S, Anandasivam A, et al. Environment-conscious scheduling of HPC applications on distributed Cloud-oriented data centers. *Journal of Parallel and Distributed Computing*, 2011, 71(6): 732-749
- [26] Wang Y, Wang X, Chen M, et al. PARTIC: Power-aware response time control for virtualized Web servers. *IEEE Transactions on Parallel and Distributed Systems*, 2011, 22(2): 323-336
- [27] Brian D, Jules W, Douglas C S. Model-driven auto-scaling of green cloud computing infrastructure. *Future Generation Computer Systems*, 2012, 28(2): 371-378
- [28] Chen Y, Iyer S, Liu X, et al. Translating service level objectives to lower level policies for multi-tier services. *Cluster Computing*, 2008, 11(3): 299-311
- [29] Wada H, Suzuki J, Yamano Y, et al. Evolutionary deployment optimization for service-oriented clouds. *Software-Practice & Experience*, 2011, 41(5): 469-493
- [30] Perez J, Germain R C, Kegl B, et al. Multi-objective reinforcement learning for responsive grids. *Journal of Grid Computing*, 2010, 8(3): 473-492
- [31] Li Qiang, Hao Qin-Fen, Xiao Li-Min, et al. Adaptive management and multi-objective optimization for virtual machine placement in cloud computing. *Chinese Journal of Computers*, 2012, 34(12): 2253-2264 (in Chinese)
(李强, 郝沁芬, 肖利民等. 云环境中虚拟机放置的自适应管理与多目标优化. *计算机学报*, 2012, 34(12): 2253-2264)
- [32] Beloglazov A, Abawajy J, Buyya R. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*, 2012, 28(5): 755-768
- [33] Ma D S, Bondade R. Enabling power-efficient DVFS operations on silicon. *IEEE Circuits and Systems Magazine*, 2011, 10(1): 14-30
- [34] Sun D W, Chang G R, Guo Q, et al. Modeling, evaluating and designing virtual machine scheduling by a clustering mechanism in cloud computing environments. *International Journal Wireless and Mobile Computing*, 2011, 5(1): 70-76
- [35] Murtazaev A, Oh S. Sercon: Server consolidation algorithm using live migration of virtual machines for green computing. *IETE Technical Review*, 2011, 28(3): 212-231
- [36] Beloglazov A, Buyya R. Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers//*Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science*. Bangalore, India, 2010: 1-6
- [37] Liao X, Hu L, Jin H. Energy optimization schemes in cluster with virtual machines. *Cluster Computing*, 2010, 13(2): 113-126
- [38] Deng Y, Pung B. Conserving disk energy in virtual machine based environments by amplifying bursts. *Computing*, 2011, 91(1): 3-21
- [39] Dhiman G, Marchetti G, Rosing T. vGreen: A system for energy-efficient management of virtual machines. *ACM Transactions on Design Automation of Electronic Systems*, 2010, 16(1): 1-27
- [40] Ye L, Lu G, Kumar S, et al. Energy-efficient storage in virtual machine environments. *ACM Sigplan Notices*, 2010, 45(7): 75-84
- [41] Mezmaza M, Melabb N, Kessacib Y, et al. A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems. *Journal of Parallel and Distributed Computing*, 2011, 71(11): 1497-1508
- [42] Maheshwari N, Nanduri R, Varma V. Dynamic energy efficient data placement and cluster reconfiguration algorithm for MapReduce framework. *Future Generation Computer Systems*, 2012, 28(1): 119-127
- [43] Wood T, Shenoy P, Ramakrishnan K K, et al. CloudNet: Dynamic pooling of cloud re-sources by live WAN migration of virtual machines. *ACM Sigplan Notices*, 2011, 46(7): 121-132
- [44] Zheng J, Ng T S E, Sripanidkulchai K. Workload-aware live storage migration for clouds. *ACM Sigplan Notices*, 2011, 46(7): 133-144
- [45] Kim K H, Beloglazov A, Buyya R. Power-aware provisioning of virtual machines for real-time Cloud services. *Concurrency and Computation: Practice & Experience*, 2011, 23(13): 1491-1505
- [46] Li C L, Chen Z L. Bin-packing problem with concave costs of bin utilization. *Naval Research Logistics*, 2006, 53(4): 298-308
- [47] Crainic T G, Perboli G, Rei W, et al. Efficient lower bounds and heuristics for the variable cost and size bin packing problem. *Computer & Operations Research*, 2011, 38(11): 1474-1482
- [48] Lewis R, Song X, Dowsland K, et al. An investigation into two bin packing problems with ordering and orientation implications. *European Journal of Operational Research*, 2011, 213(1): 52-65
- [49] Buyya R, Ranjan R, Calheiros R N. Modeling and simulation of scalable cloud computing environments and the CloudSim toolkit: Challenges and opportunities//*Proceedings of the 2009 International Conference on High Performance Computing & Simulation, HPCS 2009*. Leipzig, Germany, 2009: 1-11
- [50] Xu B, Zhao C, Hu E, et al. Job scheduling algorithm based on Berger model in cloud environment. *Advances in Engineering Software*, 2011, 42(7): 419-425
- [51] Calheiros R N, Ranjan R, Beloglazov A, et al. CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software-Practice & Experience*, 2011, 41(1): 23-50



SUN Da-Wei, born in 1985, Ph. D. . His current research interests include cloud computing, big data, trusted computing, green computing, and information security.

CHANG Gui-Ran, born in 1946, Ph. D. , professor,

Ph. D. supervisor. His current research interests include computer networks, HPC and information security.

CHEN Dong, born in 1983, Ph. D. candidate. His current research interests include Internet of Things and cloud computing.

WANG Xing-Wei, born in 1968, Ph. D. , professor, Ph. D. supervisor. His research interests are next generation Internet and mobile Internet.

Background

Green computing as a new trend for high-end computing is gaining significant momentum with global warming and ever-rising demand for more energy. In principle, cloud computing is an effective way to support green computing by providing resource consolidation with virtualization. As large-scale cloud data centers reach enormous size in terms of equipment, high energy efficiency issue is one of the major obstacles for opening up the new era of green cloud computing.

In most current cloud computing environments, the performance and service level objectives are under consideration, while high energy efficiency is ignored. This justifies the importance of profiling, quantifying, modeling, and evaluating green service level objectives in large-scale cloud data centers, so as to minimize energy consumption, to achieve adequate measurement and management of high green service level objectives in cloud computing environments.

In this paper, the definition of green cloud is given and the principles for high green service level objectives are systematically analyzed by referring to the service level objective and green computing theories. Based on the principles and

methodology of green cloud, a multi-metric energy consumption model M2EC is put forward. Theoretical as well as experimental results conclusively demonstrate that the M2EC algorithm has high potential as it provides efficient green enhancements and significant energy saving. It implements the trade-off between energy consumption and green-service level objectives efficiently and effectively in cloud computing environments.

This work is supported by the National Science Foundation for Distinguished Young Scholars of China under Grant No. 61225012; the National Natural Science Foundation of China under Grant Nos. 61070162, 71071028 and 70931001; the Specialized Research Fund of the Doctoral Program of Higher Education for the Priority Development Areas under Grant No. 20120042130003; the Specialized Research Fund for the Doctoral Program of Higher Education under Grant No. 20100042110025 and No. 20110042110024; the Specialized Development Fund for the Internet of Things from the ministry of industry and information technology of the P. R. China; the Fundamental Research Funds for the Central Universities under Grant Nos. N110204003, N120104001.