软件学报 ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*,2014,25(7):1388−1402 [doi: 10.13328/j.cnki.jos.004602]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

# 资源调度等待开销感知的虚拟机整合*

李铭夫, 毕经平, 李忠诚

(中国科学院 计算技术研究所,北京 100190)

通讯作者: 李铭夫, E-mail: limingfu@ict.ac.cn

**摘 要**: 近年来,数据中心庞大的能源开销问题引起广泛关注.虚拟化管理平台可以通过虚拟机迁移技术将虚拟机整合到更少的服务器上,从而提高数据中心能源有效性.对面向数据中心节能的虚拟机整合研究工作进行调研,并总结虚拟机整合研究存在的 3 个挑战.针对已有工作未考虑虚拟机等待资源调度带来的服务器资源额外开销这种现象,开展了资源调度等待开销感知的虚拟机整合研究.从理论和实验上证明了在具有实际意义的约束条件下,存在着虚拟机等待资源调度带来的服务器资源额外开销,且随着整合虚拟机数量的增长保持稳定.基于典型工作负载的实验结果表明,这个额外开销平均占据了 11.7%的服务器资源开销.此外,提出了资源预留整合(MRC)算法,用于改进已有的虚拟机整合算法.算法模拟实验结果表明,MRC 算法相比于常用的虚拟机整合算法 FFD(first fit decreasing),明显降低了服务器资源溢出概率.

**关键词**: 能源有效性;虚拟机整合;资源开销测量;数据中心

**中图法分类号**: TP316

中文引用格式: 李铭夫,毕经平,李忠诚.资源调度等待开销感知的虚拟机整合.软件学报,2014,25(7):1388−1402. http://www.jos.org.cn/1000-9825/4602.htm

英文引用格式: Li MF, Bi JP, Li ZC. Resource-Scheduling-Waiting-Aware virtual machine consolidation. Ruan Jian Xue Bao/ Journal of Software, 2014,25(7):1388−1402 (in Chinese). http://www.jos.org.cn/1000-9825/4602.htm

## Resource-Scheduling-Waiting-Aware Virtual Machine Consolidation

LI Ming-Fu, BI Jing-Ping, LI Zhong-Cheng

(Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100190, China)

Corresponding author: LI Ming-Fu, E-mail: limingfu@ict.ac.cn

**Abstract**: In recent years, the huge resource consumption problem of data centers is being widely concerned. Virtual machine monitor (VMM) can consolidate virtual machines (VMs) onto fewer servers via VM migration to improve the energy efficiency of data centers. This paper surveys the recent works on energy-efficient VM consolidation, and summarizes three research challenges. Among them, this work considers the resource consumption overhead caused by the waiting of virtual machines for server resource scheduling. The study theoretically and experimentally proves that under realistic constraints, this overhead remains steady as the number of consolidating VMs grows. Experiments based on a representative benchmark show that, on average, 11.7% of the server's CPU resource is occupied by the overhead. In addition, in order to fill in the gap on existing approaches, this paper proposes margin reserved consolidation (MRC) algorithm. Simulation results show that MRC outperforms the state of the art baseline in terms of server resource violation probability.

**Key words**: energy efficiency; virtual machine consolidation; resource cost measurement; data center

## 1 虚拟机整合问题的背景与意义

近年来,使用虚拟化技术的云计算服务使得企业可以随时、随地、按需地通过网络访问共享资源池的计算

* 基金项目: 国家重点基础研究发展计划(973)(2011302505); 国家自然科学基金(61070210, 61303243)
收稿时间: 2013-10-31; 修改时间: 2014-03-17; 定稿时间: 2014-05-06

资源、网络资源或存储资源,从而降低企业的 IT 运营成本,提高企业的经营效益.然而,随着云计算服务的普及,数据中心庞大的能源开销问题愈显突出:首先,数据中心能源开销规模庞大.麦肯锡的调研报告[1]指出,一个数据中心的耗电量约等于 25 000 个家庭的耗电量,且在 2010 年,全世界数据中心的电力消耗高达 110 亿美元;其次,数据中心能源开销增长迅速.报告[2]指出,从 2005 年~2010 年,全世界数据中心的能源开销提高了 56%,在 2010 年,这些数据中心占用了全球 1.1%~1.5%的电力资源,而且数据中心能源开销以每 5 年翻倍的速度增长.在这些能源开销中,数据中心的计算资源(或服务器)占据了大部分.美国能源部门的报告[3]指出,一个典型的数据中心,其服务器的能源开销占总能源开销的近 50%.然而,数据中心服务器的资源利用率却非常低下,IBM 的调研报告[4]指出,数据中心服务器的平均 CPU 使用率只有 15%~20%.这个现象恰恰为解决数据中心能源开销庞大问题提供了一条途径.

解决数据中心能源开销庞大问题的一种有效且常用的方法是虚拟机整合.虚拟机整合是指根据虚拟机的资源需求,通过虚拟机迁移将其放置到更少的服务器上,进而将部分服务器关闭或处于低功耗状态,从而减小数据中心的能源开销.在虚拟化的数据中心中,有效地整合虚拟计算资源,在满足云计算服务质量的前提下尽可能地减少服务器使用数量,可以提高数据中心的服务器资源利用率,降低能源开销,从而实现数据中心节能的要求.当前广泛使用的虚拟机管理平台,如 VMware[5],Xen[6]等,都支持在同构的服务器之间进行虚拟机实时迁移,实现不同服务器上的虚拟机整合.VMware 的报告[7]指出,通过虚拟机整合,可以降低约 50%的硬件和运营开销以及 80%的能源开销,从而每台虚拟化服务器每年可节约 3 000 美元.

虚拟机整合问题的两个优化目标是:最小化数据中心能源开销以及保证云计算服务质量.数据中心能源开销通常用服务器使用数量衡量,因为服务器占用了绝大部分的能源开销,并且能源开销与服务器数量呈正相关;服务质量则用云计算服务级别协议(service level agreement,简称 SLA)中的一系列指标来定义.因为这两个优化目标是权衡(trade-off)关系,因此,虚拟机整合问题往往转化为在特定的服务质量约束条件下,最小化能源开销的问题.

虚拟机整合问题具有一定的复杂性,该研究问题的两个挑战是:首先,其中的虚拟机放置问题是一个 NP 困难的优化问题.虚拟机放置是指给定一个虚拟机集合和一个服务器集合,在满足服务器资源开销约束条件下,将虚拟机放置到最少的服务器中.这个过程常常被抽象为装箱问题[8],装箱问题的解决具有 NP 困难的计算复杂性,在实际应用中,常使用启发式算法来解决,而启发式算法无法保证找到全局最优解.此外,实际的数据中心为虚拟机整合问题带来诸多影响因素.例如,虚拟机迁移和整合过程产生的服务器资源开销、虚拟机不同的工作负载具有不同的资源需求、服务器和虚拟机的异构性对虚拟机整合算法的影响以及云计算数据中心的庞大规模对虚拟机整合算法效率的影响.

学术界对面向数据中心节能的虚拟机整合问题展开了深入研究,但是我们经过对已有研究工作的调研和总结(详见本文第 2 节)发现,已有工作存在一个共同的不足之处:已有工作在进行虚拟机整合时,没有考虑虚拟机等待服务器资源调度带来的服务器 CPU 资源的额外开销.为了提高服务器资源利用率,一个服务器通常运行多个虚拟机,当虚拟 CPU 数量大于物理 CPU 数量时,总有一部分虚拟机需要等待服务器调度 CPU 资源,这个等待时间将计算到服务器的 CPU 使用率中(例如 Linux 系统 CPU 使用率的 Steal Time 字段),产生服务器 CPU 资源的额外开销.这个额外开销,使得运行虚拟机实际所需的服务器数量大于虚拟机整合算法的预期结果.如果使用已有工作算法,服务器的 CPU 资源将面临过高负载风险,从而降低虚拟机的应用性能,最终降低云计算的服务质量.

基于已有工作的不足之处,本文研究了资源调度等待开销感知的虚拟机整合.在本文中,我们以云计算操作系统 Xen Cloud Platform(XCP)[6]的 CPU 资源调度机制为研究对象.XCP 是用于建设云计算环境的开源系统,使用了基于配额(credit-based)的机制为每个虚拟机调度 CPU 资源.我们在文中将 XCP 调度 CPU 资源的过程模拟成 $M/M/n$ 队列系统,在现实约束条件下,证明了虚拟机等待服务器资源调度带来的服务器 CPU 资源额外开销是存在且保持稳定的,与服务器上的虚拟机数量无关;然后,基于理论分析结果,我们提出了资源预留整合(margin reserved consolidation,简称 MRC)算法以弥补已有工作的不足;最后,我们在 XCP 上进行了 3 个典型数据中心工

作负载的资源开销实验,实验结果显示:虚拟机等待服务器资源调度带来的服务器 CPU 资源额外开销占服务器 CPU 资源开销的 10%~30%,对于运行 4~12 个虚拟机的服务器来说,相当于平均每个服务器有 1.2 个虚拟机完全得不到任何资源.

本文的主要贡献如下:

1) 本文对近 5 年的面向数据中心节能的虚拟机整合研究工作进行调研及总结,并提出虚拟机整合研究存在的 3 个挑战;
2) 据我们所知,本文首次对虚拟机等待资源调度带来的服务器 CPU 资源额外开销进行了建模分析,并基于数据中心典型工作负载实验验证了该额外开销的分析结果;
3) 本文在考虑上述额外开销的基础上,提出了资源预留整合算法,弥补了已有工作的不足.该算法与常用虚拟机整合算法相比,明显降低了服务器溢出概率.

## 2 面向数据中心节能的虚拟机整合相关工作总结

虚拟机整合问题研究的需求来源于提高数据中心能源有效性,由于近年来虚拟化技术和云计算的发展,这个问题在学术界越来越受关注.本节对近 5 年面向节能的虚拟机整合策略研究的已有工作进行总结,按其研究侧重点,可将已有工作分为 4 类:资源开销感知的虚拟机整合;性能感知的虚拟机整合;网络感知的虚拟机整合以及工作负载感知的虚拟机整合.对于每一类研究工作,根据其使用方法或考虑角度的不同,又分为若干子类.已有工作分类结果见表 1.

**Table 1** Summary of research works on efficiency-aware virtual machine consolidation in data centers

**表 1** 面向数据中心节能的虚拟机整合研究工作分类

| | | |
|---|---|---|
| 资源开销感知<br>(cost-aware) | 基于装箱模型算法<br>虚拟机迁移资源开销感知 | 文献[10–14]<br>文献[15–21] |
| 性能感知<br>(performance-aware) | 从云服务用户角度<br>从云服务管理者角度 | 文献[22–27]<br>文献[28–34] |
| 网络感知<br>(network-aware) | 基于虚拟机网络带宽需求<br>基于虚拟机流量关系<br>基于网络拓扑结构 | 文献[35–45]<br>文献[46–50]<br>文献[51–56] |
| 工作负载感知<br>(workload-aware) | 基于工作负载特征<br>面向特定应用 | 文献[57,58]<br>文献[59–64] |

### 2.1 资源开销感知(cost-aware)的虚拟机整合

资源开销感知的资源是指服务器资源.资源开销感知的虚拟机整合研究分为两类:一类研究工作将虚拟机整合问题抽象为装箱问题,以服务器的各项资源作为箱子的维度,提出启发式算法最小化箱子(服务器)的使用数量;另一类研究工作则重点研究虚拟机迁移过程产生的服务器资源开销,及其对虚拟机整合策略的影响.

#### 2.1.1 基于装箱模型算法

Srikantaiah 等人[9]的工作是最早将虚拟机整合问题抽象为装箱问题的研究工作,他们在实验中发现:在服务器的 CPU 使用率和硬盘使用率之间存在一个最优化组合,可以使得服务器的能源开销最小.Cardosa 等人[10]在进行虚拟机整合时,充分利用了虚拟化管理平台的 min,max 以及 share 这 3 个参数,这些参数分别代表分配到某个虚拟机的最小、最大及共享部分的 CPU 资源,从而显著提高了服务器的资源利用率.Feller 等人[11]考虑了多种服务器资源,将问题抽象为多维装箱问题,并首次使用了人工智能方法(蚁群优化算法)用于虚拟机放置算法.He 等人[12]则考虑了多个虚拟机形成虚拟集群的场景,在保证虚拟集群满足服务质量(quality of service,简称 QoS)的情况下,认为集群中虚拟机的资源需求是可塑的.Ghribi 等人[13]同时考虑了虚拟机放置与虚拟机迁移,使用装箱模型来解决此联合问题,以同时最小化能源开销和迁移开销.

#### 2.1.2 虚拟机迁移资源开销感知

虚拟机整合是通过虚拟机迁移来实现的,因此,已有研究工作大多关注虚拟机迁移带来的资源开销.

Voorsluys 等人[14]通过实验测量了 Xen 虚拟机实时迁移对其应用程序响应时间的影响,实验结果显示,迁移开销是可接受但不可忽略的.Akoush 等人[15]提出了迁移模拟模型用于预测 Xen 虚拟机迁移所需的时间,虚拟机迁移时间也是影响 SLA(service level agreement)评价指标的重要因素.Ye 等人[16]发现,虚拟机实时迁移对服务器性能的影响与服务器的内存、CPU 及工作负载有关,他们提出服务器资源预留机制用于提高虚拟机整合过程中的迁移效率.Huang 等人[17]预测了虚拟机实时迁移对服务器能源开销的影响,并发现服务器能源开销与 CPU 使用率呈线性关系.与之不同的是,Strunk 等人[18]发现,虚拟机实时迁移对服务器的能源开销与虚拟机占用内存及可用带宽大小有关.以上提到的研究工作重点在于测量和预测虚拟机迁移的资源开销,Liu 等人[19]和 Setzer 等人[20]的工作则在虚拟机整合过程中考虑了这种开销.Liu 等人[19]同时研究虚拟机实时迁移对服务器性能和服务器能源消耗的影响,他们提出一种机制来预测这种影响,并基于该机制提出虚拟机整合策略,显著降低了虚拟机迁移的性能开销和服务器的能源开销.Setzer 等人[20]预测了虚拟机迁移过程所产生的资源需求,并在虚拟机位置随时间动态变化的过程中考虑了这种资源需求.

### 2.2 性能感知(performance-aware)的虚拟机整合

性能感知的虚拟机整合研究分为两类:一类研究工作从云服务用户的角度,以提高云服务的服务质量(QoS)或避免服务级别协议(SLA)违规为目的;另一类研究工作从数据中心或云服务管理者的角度,以提高数据中心的服务能力或收益为目的.

#### 2.2.1 从云服务用户角度

云服务的 QoS 通常用 SLA 中的一系列指标来衡量,如网络吞吐量或服务响应时间.以此为优化目标的代表性研究工作是 Beloglazov 等人的工作[21−23],其相同点是基于虚拟机的 CPU 使用率决定服务器上哪些虚拟机需要被迁移,从而避免服务器资源过载带来 SLA 违规.在文献[21]中,他们提出一种基于自适应 CPU 使用率阈值的虚拟机动态整合机制,虚拟机的 CPU 使用率通过历史数据的统计分析来预测;文献[22]并非通过历史数据来预测虚拟机的 CPU 使用率,而是使用 3 种不同的策略来人工设定阈值上限和阈值下限.这 3 种策略分别是:最小虚拟机迁移数量、最小服务器 CPU 使用率增长以及随机策略.上述两个工作使用了启发式方法,可能无法找到全局最优结果,文献[23]则使用了马尔可夫链模型并提出一种控制算法,在特定的 QoS 约束条件下最大化每两次虚拟机迁移的时间,从而决定哪些服务器是资源过载的.Dong 等人[24]则采用了更复杂和更精确的方法来预测虚拟机的 CPU 使用率.他们为每个虚拟机建立了自己的资源需求数据模型,并使用了一种分布式决策支持系统(rDSS)作为云计算应用来预测虚拟机的 CPU 使用率.Berral 等人[25]的工作考虑的 SLA 指标是应用程序的运行时间,他们提出一种自动化机制来进行任务调度,通过机器学习的方法来估计目标优化方程中的参数.Wang 等人[26]以数据库应用作为研究案例,以数据库系统的 CPU 和 I/O 资源利用率作为优化目标,基于 Fuzzy 模型提出了一种虚拟机资源管理机制,通过虚拟机和主机之间的相互通信,协同优化资源调度过程和应用性能.

#### 2.2.2 从云服务管理者角度

从数据中心或云服务管理者的角度出发,Feller 等人[27]基于服务器之间的 P2P 网络提出一种非集中式的动态虚拟机整合机制,通过周期性地交互服务器信息,提高虚拟机整合算法的收敛速度.Maguluri 等人[28]将用户对虚拟机的请求模拟为随机过程,采用基于时间帧的非抢占式虚拟机分配策略,优化虚拟机请求过程的效率并提高服务器负载均衡.Xiao 等人[29]则利用“偏斜度”来衡量服务器资源使用的不均衡性,通过最小化服务器的“偏斜度”来提高服务器的资源利用率.Xu 等人[30]考虑了资源调度的公平性,通过获取用户对虚拟机资源的需求,使用一种多对一的稳定匹配方法,有效地将虚拟机映射到服务器上,解决不同用户的请求冲突.Zhang 等人[31]考虑的则是资源调度的时延,他们提出一种动态资源调度机制,根据服务器的能源开销和动态重配置开销计算所需服务器的数量,以在能源节约和资源调度时延之间取得较好的均衡.除了以提高数据中心服务提供能力为目的外,还有一部分研究工作以数据中心整体收益作为优化目标.Shi 等人[32]的工作考虑将虚拟机放置到给定的一个服务器集合时,最大化 IaaS 服务提供商的收益.他们将问题抽象为多维装箱问题,给每种资源使用赋予一个收益权重,并采用启发式方法最大化总体的资源使用收益.Zheng 等人[33]的工作则是考虑在虚拟机需要进行重新调度时,给定数据中心的可用资源和虚拟机重新调度的时限,最大化数据中心的收益.同样,他们也是为服务器

的每种资源的使用赋予一个收益权重,并最大化整体收益.

### 2.3 网络感知(network-aware)的虚拟机整合

网络感知的虚拟机整合研究旨在同时最小化数据中心能源开销与数据中心通信流量,其研究工作根据研究方法的不同可分为 3 类:第 1 类研究工作基于虚拟机对网络带宽的需求;第 2 类研究工作基于虚拟机的流量关系;第 3 类研究工作则基于网络拓扑结构.

#### 2.3.1 基于虚拟机对网络带宽的需求

Kliazovich 等人[34]较早地将数据中心能源开销与工作流带宽需求同时加以研究,他们提出一种面向数据密集型工作负载的任务调度机制,在最小化服务器使用数量的同时,避免网络出现流量热点.Sonnek 等人[35]使用了自定义协议用于交换虚拟机的信息,为最优化的虚拟机迁移策略提供参考.他们的方法是监测虚拟机的流量需求模式,并使用分布式交互算法来动态适应虚拟机调度,目的在于最小化虚拟机的通信流量开销.

Wang 等人[36]认为,虚拟机的网络带宽需求是服从正态分布的随机变量,并将虚拟机整合问题抽象为随机装箱问题.他们提出的在线算法相对于已有工作可以减少 30%的箱子数量.Breitgand 等人[37]采用了与文献[36]相同的假设和模型,通过引入带宽冲突风险控制,进一步提高了在线算法的性能.Biran 等人[38]的研究工作旨在同时满足虚拟机的带宽需求和 CPU 及内存需求,他们将虚拟机整合问题抽象成权重感知的最小分割问题,将网络划分为两个非空子集,并提出启发式算法以最小化网络分割负载比例的最大值.Dong 等人[39]虽然也通过使用装箱模型来优化各种服务器资源的开销,但与上述其他工作的不同之处在于,他们的优化目标是同时最小化服务器数量和网络设备数量,从而更好地实现节能.

Ghorbani 等人[40]则从另一个角度考虑此问题,认为虚拟机迁移会引起网络状态变化,从而改变网络可提供带宽的大小.他们基于 SDN 同时规划虚拟机放置序列和网络状态改变序列来解决此问题.Mann 等人[41]也是基于 SDN 的方法,在 OpenFlow 网络中考虑到虚拟机的带宽需求是随时间变化的,从而提出一种虚拟机管理机制,根据迁移开销和可用带宽选择合适的迁移目的服务器.

Wang 等人[42]研究了虚拟机带宽需求随时间变化的具体特征,他们通过分析 Wikipedia 数据中心的数据,发现超过 80%的工作负载在时间上是不相关的,不会同时达到流量峰值.基于该观察结果,他们将时分复用思想应用到整合策略中,以提高能源节省效率.Cohen 等人[43]认为,对于带宽需求密集型的应用来说,由于同一条链路被多个虚拟机和多个服务器共享,优化问题变得复杂.因此,他们通过选择一个根节点并优化虚拟机发往根节点的流量开销来简化该优化问题.Popa 等人[44]也认同该优化问题的复杂性,他们首先研究了用户最小网络带宽保证、网络负载均衡和带宽利用率三者之间的权衡关系,并使用一系列的资源调度策略来找到这些权衡关系的最优平衡点.

#### 2.3.2 基于虚拟机的流量关系

基于虚拟机流量关系的虚拟机整合策略的核心思想是将具有流量关系的虚拟机尽量近地整合在一起,例如在同一个服务器或同一个机架中.Meng 等人[45]通过真实数据中心的数据研究虚拟机流量矩阵的特征,发现虚拟机的流量大小分布是不均衡的,且每个虚拟机的流量在长时间内是稳定的.因此,他们提出启发式算法将具有较大流量关系的虚拟机放置在相近位置,最小化高层交换机的负载,从而提高数据中心网络的可扩展性.Shrivastava 等人[46]针对多层次企业级应用,研究虚拟机的流量依赖性,并结合网络拓扑信息,在最小化数据中心流量的同时满足服务器的资源开销约束条件.Dias 等人[47]的工作与文献[46]相似,通过收集虚拟机的流量信息和网络拓扑信息,并对服务器和虚拟机进行分组,将具有流量关系的虚拟机尽量地整合在一起.Zhang 等人[48]的工作也是采用相似算法来最小化数据中心的流量,但他们假设的应用场景是虚拟机与服务器的资源开销特征是具有先验知识的.Hu 等人[49]的优化目标则是数据中心网络的对分带宽(或对半带宽,bi-section bandwidth),以运行同一个应用程序的具有流量关系的虚拟机集合作为调度对象,旨在提高数据中心网络的整体通信能力.

#### 2.3.3 基于网络拓扑结构

Stage 等人[50]在虚拟机整合过程中,同时考虑了虚拟机迁移的网络带宽开销以及典型的树状数据中心网络拓扑结构,他们的工作是基于网络拓扑方法的较早工作.Alicherry 等人[51]考虑的是多个数据中心的分布式云计

算环境,为了尽量减小跨数据中心的数据访问延时,他们在虚拟机放置策略中最小化所选数据中心的最大距离.在数据中心内部也采用相似方法,考虑的也是典型的树状数据中心网络拓扑.Jain 等人[52]研究的是多个根节点的树状数据中心网络拓扑问题,通过减少高负载服务器数量的方法来同时满足服务器资源开销和链路带宽利用.Giurgiu 等人[53]研究的则是大规模数据中心网络结构下的虚拟机整合,他们以虚拟网络架构作为放置对象,在调度过程中综合考虑了计算资源、网络资源和网络访问性能.Jiang 等人[54]研究了虚拟机放置与虚拟机路由联合问题,在算法中同时考虑链路带宽开销和服务器开销.他们使用了 Markov 模型并提出了在线算法,并在 Fat-Tree 结构的网络拓扑中进行算法性能评测.Yang 等人[55]考虑的则是大规模异构数据中心网络下的虚拟机放置问题,基于 Shadow-Routing 方法,提出虚拟机路由与虚拟机放置的联合优化算法,最小化多个数据中心的最大平均资源使用率.

### 2.4 工作负载感知(workload-aware)的虚拟机整合

上述研究工作中经常提及的一个现象是虚拟机整合策略的性能和效率与虚拟机上的工作负载有关,因此,学术界针对数据中心的典型工作负载研究虚拟机整合策略.工作负载感知的虚拟机整合研究分为两类:一类基于工作负载的特征来进行研究;另一类则面向特定的应用,如 MapReduce 或高性能计算.

#### 2.4.1 基于工作负载特征

Yang 等人[56]将虚拟机的工作负载分为数据密集型和计算密集型两种,在实验中发现:服务器有无本地虚拟机镜像,对数据密集型应用的性能影响远大于对计算密集型应用的性能影响.基于此实验观察,他们在虚拟机整合算法中根据工作负载类型进行服务器选择,在最小化能源开销的同时最大化应用性能.Tan 等人[57]则重点研究工作负载的特征预测能为虚拟机整合的节能效果带来多大的收益,他们在实验中找到了一个最优的预测窗口大小,并将其应用到虚拟机整合算法中.

#### 2.4.2 面向特定应用

由于 MapReduce 在数据中心得到了广泛应用,因此学术界的研究成果主要面向 MapReduce 应用.Palanisamy 等人[58]在进行虚拟机放置时考虑了数据本地性,有效地降低了 MapReduce 应用的运行时间,并且减少了网络的通信流量.Huang 等人[59]则同时考虑数据中心中存在 MapReduce 应用的虚拟机和非 MapReduce 应用的虚拟机,利用整数非线性优化模型来研究该虚拟机整合问题.考虑到实际应用中,虚拟机调度总有一个时限(dead-line),Hwang 等人[60]研究的 MapReduce 应用虚拟机调度问题加入了调度时限这一约束条件,其算法目的在于最小化用户使用虚拟机的开销.Li 等人[61]也从用户角度提出一种资源调度机制,根据服务器的存储使用率、CPU 负载和链路带宽,增强 MapReduce 应用的数据本地性和计算节点(虚拟机)本地性,并且在应用运行过程中自适应地进行资源重新调度,从而提高 MapReduce 的应用性能.Alicherry 等人[62]认为,在 MapReduce/Hadoop 这种数据密集型应用中,从处理器节点到数据节点的访问延迟是影响应用运行时间的最关键因素.因此,他们通过优化虚拟机放置来解决此问题,提出了优化算法,在满足各种约束条件下找到虚拟机的最优放置位置,并解决了最小化时延和带宽开销之间的权衡问题.Rodero 等人[63]研究的则是面向高性能计算(HPC)应用的虚拟机整合问题,在他们所考虑的应用场景中,服务器具有可配置性.因此,他们通过应用聚类将资源利用相似的应用整合在同组服务器中,通过细粒度的服务器资源控制达到更好的节能效果.

### 2.5 虚拟机整合策略研究存在的挑战

通过对已有研究工作的调研分析我们发现,面向数据中心节能的虚拟机整合研究工作仍存在以下挑战:

1) 由于虚拟机整合是 NP 困难的优化问题,已有研究工作采用启发式算法为虚拟机寻找满足条件的服务器.然而,实际数据中心的网络拓扑结构以及服务器的部署策略往往是可知的,根据网络拓扑信息和服务器位置信息,可以优化启发式算法寻找最优解的过程,提高算法的性能和收敛效率;
2) 实际的数据中心总是以循序渐进的方式建设,往往存在多种架构的服务器.另外,为了避免单一厂商的束缚(vendor lock-in)以及节省采购开销,往往采用多种虚拟化技术,如同时采用商用产品 VMware 和开源产品 KVM 等.在虚拟机整合过程中,需要考虑异构服务器、异构虚拟化技术以及不同工作负

载对服务器资源开销以及整个数据中心能源开销的影响;

3) 虚拟机整合的一个重要步骤是判断服务器是否能够容纳运行在其上的虚拟机,这个判断不仅要考虑虚拟机的资源需求之和与服务器资源容量的关系,还需考虑虚拟机等待资源调度或服务器虚拟交换机流量等因素带来的服务器资源的额外开销.

本文主要考虑上述的第 3 个挑战,研究资源调度等待开销感知的虚拟机整合问题.

## 3 问题描述及算法

虚拟机整合问题的研究与解决往往使用装箱模型,本节首先给出多维随机装箱模型(multi-dimensional stochastic bin packing)的问题描述;然后,对整合过程中虚拟机等待资源调度带来的服务器资源额外开销问题进行建模,介绍 XCP 基于配额的 CPU 调度机制及该机制下服务器资源额外开销的理论分析结果;最后,基于装箱问题的启发式算法 First Fit Decreasing(FFD),给出我们的资源预留整合算法.

### 3.1 多维随机装箱模型问题描述

虚拟机及运行虚拟机的服务器分别作为装箱模型中的物件(item)与箱子(bin),物件及箱子是多维的,每个维度代表一种资源(如 CPU、内存或网络带宽等).物件的维度大小表示其相应的资源需求,而箱子的维度大小表示其相应的资源容量.考虑到虚拟机的资源需求往往是动态变化的,因此,物件的维度大小是随机变量.

假设有 $N$ 个虚拟机需要装进服务器中,并且资源维度为 $R$.令随机变量 $x_i^r$ 表示第 $i$ 个虚拟机的第 $r$ 项资源的资源需求.该问题的目标在于:最小化可以装入所有虚拟机的服务器数量,并且满足服务器资源溢出概率小于一个给定的常数 $p$.服务器溢出概率是指服务器上所有虚拟机的某项资源需求之和超过服务器资源容量的概率.$p$ 是一个(0,1)的正常数,可以由云计算服务级别协议 SLA(service level agreement)推断得出,是云计算服务质量的重要衡量指标.

多维随机装箱模型描述如下:

给定一种 $R$ 维度的箱子 $S$,箱子的维度大小为$(c^1,c^2,\ldots,c^R)$,给定 $N$ 个 $R$ 维度的物件,其中第 $i$ 个物件的维度大小为 $(x_i^1,x_i^2,\ldots,x_i^R)$,找到一个最小箱子数量 $B$,用于放置所有物件,并且对于每个箱子,满足服务器溢出概率小于 $p\in(0,1)$.

### 3.2 基于配额的CPU调度机制

如上所述,为了使服务器使用数量最小,往往一个服务器上运行多个虚拟机.考虑到内存及硬盘资源在虚拟机建立时就预先分配好,我们在本文中只考虑 CPU 资源.当虚拟 CPU 数量大于物理 CPU 数量时,总有虚拟 CPU 需要等待服务器 CPU 调度器的资源调度,本文的核心问题就是研究虚拟 CPU 等待资源调度对虚拟机整合过程的影响.

基于配额(credit-based)的 CPU 调度器[64]是 XCP 默认的调度机制.相比于 BVT(borrowed virtual time)调度机制[65]和简单最小时限调度(simple earliest deadline first,简称 SEDF)机制[65],基于配额的调度机制[66]在调度多处理器和 QoS 控制上表现的更好,因此,我们研究该调度机制下的 VCPU 资源调度等待开销问题.在运行 XCP 操作系统的服务器上,每一个物理 CPU(PCPU)维持一个先进先出(first in first out)的虚拟 CPU(VCPU)队列.为了保证调度公平性,赋予 VCPU 一个预先设定的配额值.每个在队列中的 VCPU 可能有两种状态:over 和 under.两种状态分别表示在当前一次资源调度周期内,VCPU 是否用完了自己的配额.VCPU 运行时会消耗配额值,如果配额值为负,那么它的状态是 over;如果配额值为正,那么它的状态是 under.每个 PCPU 在做调度决策时,会首先考虑位于队首的处于 under 状态的 VCPU.如果 PCPU 在自己队列中找不到处于 under 状态的 VCPU,则去其他队列找,以此机制来实现 CPU 资源调度的公平性.

在本文中,我们使用 XCP 基于配额的 CPU 调度器研究 VCPU 等待 PCPU 资源调度引起的服务器资源额外开销问题,但是这种额外开销及其对虚拟机整合策略的影响同样出现在其他虚拟化平台上,例如 KVM.KVM 使用 QEMU 模拟处理器,其 VCPU 调度机制本质上是 Linux 的线程调度机制,Linux 系统使用的优先级调度模型

与 XCP 的基于配额的调度机制有相似之处,因此,本文的研究结果对研究 KVM 平台的虚拟机整合策略有借鉴作用.在 KVM 等其他虚拟化平台上的资源调度等待开销感知的虚拟机整合策略研究,是我们下一步的研究工作.

### 3.3 服务器资源额外开销分析

基于上述的 CPU 调度机制,我们可以不失一般性地假设每个虚拟机有一个 VCPU,每个虚拟机的 CPU 使用率是独立的,且某一服务器上虚拟机(或 VCPU)的数量在一个长时间内是随机变量.由此,我们得出定理 1.

**定理 1**. 服务器上 VCPU 的数量是一个服从泊松分布的随机变量.

证明:设给定某个时间 $t$,VCPU 的数量是 $N(t)$,显然 $N(0)=0$.首先,我们证明 VCPU 到达这一随机过程是独立增量过程.给定 $0<t_1<t_2\leqslant t_3<t_4$,$N(t_2)-N(t_1)$和 $N(t_4)-N(t_3)$是相互独立的;然后,我们证明这一随机过程是稳定增量过程.给定时间点 $s$ 和 $t_1$,$N(t_1+s)-N(t_1)$与 $s$ 有关而与 $t_1$ 无关;最后,给定时间区间$(t+\Delta t)$,有一个 VCPU 到达的概率是 $\lambda t+o(\Delta t)$,其中,$\Delta t\to 0$,$\lambda$是表示 VCPU 平均到达概率的常量.而有一个以上 VCPU 到达的概率是 $o(\Delta t)$.综上所述,由泊松分布定义,我们可以得到服务器上 VCPU 到达过程是一个泊松过程,因此,VCPU 的数量是一个服从泊松分布的随机变量. □

基于第 3.2 节的介绍,我们可以将 CPU 调度器抽象为一个有多个服务窗口和一个 FIFO 队列的排队系统.因此,我们得出定理 2.

**定理 2**. XCP 的 CPU 资源调度系统是一个 $M/M/n$ 排队系统.

证明:设 $n$ 是排队系统中的服务窗口数量(即 PCPU 数量),$k$ 是排队系统的客户数量(即 VCPU 数量).根据定理 1,我们设 VCPU 到达过程是一个平均到达率为$\lambda$的泊松过程.不失一般性,我们假设所有 PCPU 为 VCPU 服务的过程是相互独立的,对每一个 PCPU,其服务时间服从平均服务率为$\mu$的负指数分布.所以,整个排队系统的平均服务率为 $n\mu$.综上所述,由 $M/M/n$ 排队系统定义可知,XCP 的 CPU 资源调度系统是一个 $M/M/n$ 排队系统.

对于 $M/M/n$ 排队系统,学术界已经证明[67]:在满足 $\frac{\lambda}{n\mu}<1$ 的条件下,$M/M/n$ 排队系统具有稳定分布.这意味着,系统每个状态的概率是确定的.每一个客户在 $M/M/n$ 排队系统中的平均等待时间是 $W=\frac{\rho_1^n p_0}{\mu n\cdot n!(1-\rho)^2}$.其中,$\rho_1=\frac{\lambda}{\mu},\rho=\frac{\lambda}{n\mu}$.系统有 0 个客户的概率 $p_0=\left(\sum_{k=0}^{n-1}\frac{\rho_1^k}{k!}+\frac{\rho_1^n}{n!}\frac{1}{1-\rho}\right)$. □

由此,我们得出定理 3.

**定理 3**. 在满足约束条件 $\frac{\lambda}{n\mu}<1$ 以及 $k\geqslant n$ 的情况下,虚拟机等待资源调度带来的服务器 CPU 资源额外开销存在且保持稳定,并与虚拟机数量无关.

证明:我们首先说明约束条件 $\frac{\lambda}{n\mu}<1$ 和 $k\geqslant n$ 是具有实际意义的.根据第 1 节的背景介绍,大多数数据中心服务器的 CPU 使用率较低,意味着 PCPU 总有空闲资源可以分配给 VCPU.因此,排队系统的服务率总大于客户达到率,即 $n\mu>\lambda$.另外,Xen 等虚拟化平台支持在服务器上运行多个虚拟机,因此,VCPU 的数量 $k$ 总是大于 PCPU 的数量 $n$(一般为 4 或 8).因此,考虑这两个现实约束条件,一个 VCPU 平均等待资源调度的时间为

$$W=\frac{\rho_1^{2n}}{\mu n\cdot(n!)^2(1-\rho)^3}.$$

该平均等待时间与 VCPU 的到达率$\lambda$及 PCPU 的服务率$\mu$有关,与 VCPU 的数量无关.在 $M/M/n$ 排队系统中,$\lambda$和$\mu$都是常数.第 1 节的背景介绍提到,Linux 操作系统 CPU 使用率的“Steal Time”字段衡量 VCPU 的等待时间,VCPU 的等待时间将直接计算到服务器的 CPU 使用率中,这部分 CPU 使用率即我们提到的 CPU 资源额外开销.因此,服务器 CPU 资源额外开销与等待时间呈线性关系,可用公式 $O=C\cdot W$ 表示,其中,$C$ 是一个常数.

综上所述,在现实约束条件下,虚拟机等待资源调度带来的服务器 CPU 资源额外开销存在且保持稳定,并与

虚拟机数量无关. □

### 3.4 资源预留整合算法

虚拟机整合问题常被抽象为装箱问题,以每项资源(如 CPU、内存或网络带宽)作为箱子和物件的维度.已有研究工作的虚拟机整合算法没有考虑上述提到的服务器 CPU 资源额外开销.FFD(first fit decreasing)是常用的装箱问题启发式算法,是已有的虚拟机整合算法的基本策略.已有算法没有考虑文中提及的资源调度等待开销,将会低估放置所有虚拟机所用的服务器数量,这带来的结果是服务器溢出概率上升.如果服务器溢出概率超出了 SLA 规定的概率值,则意味着虚拟机性能下降,最终降低了云计算的服务质量.

为了弥补已有工作的不足,我们基于常见的装箱问题启发式算法 FFD 提出资源预留算法(margin reserved consolidation,简称 MRC).在算法中,首先根据虚拟机的 CPU 使用率进行降序排序;然后,为每个服务器预留一定的 CPU 资源;最后,以 First Fit 策略将虚拟机放置到服务器中.算法中的 marginReserved 函数为服务器预留资源,定理 3 的公式可以辅助计算该函数具体为服务器预留多少资源.但一个更有效的办法是提前测量服务器的 CPU 资源额外开销.在下一节,我们将研究一些典型工作负载下的 CPU 资源额外开销问题.资源预留算法伪代码如下所示:

```
1. Input: vmList, hostList;          Output: AllocatingDecision.
2. vmList.sortDecreasingUtilization(·);
3. hostList.marginReserved(·);
4. foreach vm in vmList do
5.     minResource←MAX
6.     allocatedHost←NULL
7.     foreach host in hostList do
8.         if host has enough resource for vm then
9.             resource=estimate(host,vm)
10.        if resource<minResource then
11.            allocatedHost←host
12.            minResource←resource
13.    if allocatedHost≠NULL then
14.        allocate vm to allocateHost
15. return AllocatingDecision
```

## 4 服务器资源额外开销测量实验

为了量化虚拟机等待资源调度带来的服务器资源额外开销,我们基于 XCP 云计算操作系统进行了典型工作负载的服务器资源开销测量实验.本节首先介绍实验环境设置,然后分析实验结果.

### 4.1 实验设置

我们的所有实验均在曙光 A620r-F 服务器上运行,该服务器有 2 个双核 AMD Opteron(tm) 2GHz 处理器,16GB 内存,5 个 127GB SATA 硬盘以及 2 个 1Gbps 以太网口.服务器运行 Xen Cloud Platform 1.1 操作系统,所有虚拟机运行 Cent OS 5.3 操作系统.服务器有 4 个物理 CPU(PCPU),且为每个虚拟机配置 4 个虚拟 CPU(VCPU),512MB 内存以及 12GB 硬盘空间.

对于每个工作负载,我们改变服务器上的虚拟机数量,并测量服务器的 CPU 使用率以及虚拟机的 CPU 使用率之和,以两者之差计算服务器 CPU 资源额外开销.本实验所展示的实验结果是 5 次实验的平均值,数据的最大值和最小值则以误差线形式展现.本实验采用了 3 个典型数据中心工作负载,包括:

1) Linux Idle:空负载的 Linux 操作系统,模拟虚拟机空负载情况;

2) Fhourstone[68]:这是一个计算 connect-4 程序结果的基准程序,我们使用 cpulimit[69]工具限制 Fhourstone 程序的 CPU 使用率,以模拟固定 CPU 负载的虚拟机;
3) LMbench[70]:这是一套系统性能测试基准程序,包括缓存文件读取、内存操作、文件系统创建与删除等.我们使用 LMbench 来模拟运行高性能应用程序的虚拟机.

### 4.2 实验结果

图 1 展示了虚拟机空负载(Linux idle)情况下服务器的 CPU 使用率,虚拟机数量由 1~24 递增.从图中可以看出:即使是空负载,虚拟机仍需占用一定的服务器 CPU 资源,且该 CPU 资源开销与虚拟机数量基本上呈线性关系.这是因为服务器需要资源维护虚拟化模块,例如为每个虚拟机维护虚拟 CPU 和虚拟网卡.注意,图 1 展示的服务器 CPU 资源开销并非由虚拟机等待资源调度引起的,而是服务器使用虚拟化技术本身带来的额外开销.因此,当我们计算 Fhourstone 和 LMbench 工作负载下的 CPU 额外开销时,需减去虚拟机空负载情况下的服务器 CPU 资源开销.

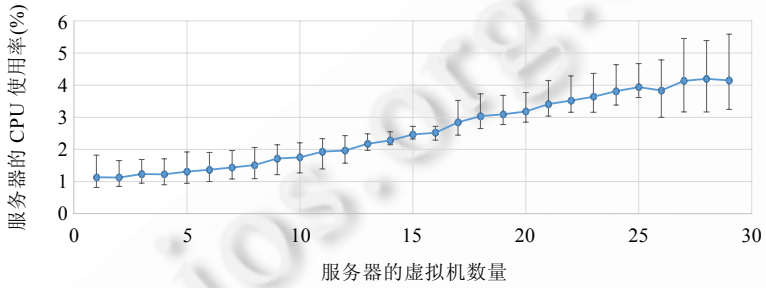


Fig.1 Server CPU resource consumption with idle virtual machine workload
图 1 虚拟机空负载下服务器 CPU 资源开销

表 2 展示了虚拟机运行 Fhourstone 工作负载时服务器的 CPU 使用率、资源调度等待引起的服务器 CPU 额外开销以及该额外开销占整个服务器 CPU 使用率的比例.考虑定理 3 中的约束条件,虚拟机数量由 4~12 递增.为了使服务器尽量多地运行虚拟机,我们使用 cpulimit 工具将每个虚拟机的 CPU 使用率控制在 5%左右.从实验结果可知:服务器 CPU 额外开销约为 6.2%,且与虚拟机数量的增长无关.此外,这个额外开销平均占服务器 CPU 总开销的 11.7%.我们认为,该额外开销对于虚拟机整合的影响是显著的,体现在如下两个方面:(1) 每个虚拟机的 CPU 使用率约为 5%,而服务器额外开销达 6.2%.意味着当服务器处于高 CPU 负载的情况下,若不为服务器预留资源,相当于平均每个服务器有 1.2 个虚拟机完全得不到任何资源,其结果是虚拟机应用性能的下降;(2) 服务器 CPU 额外开销占服务器 CPU 总开销的 11.7%,对服务器资源利用率有较大影响.

**Table 2** Server CPU resource consumption with Fhourstone workload
**表 2** Fhourstone 工作负载下服务器 CPU 资源开销

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 服务器 CPU 额外开销(%) | 6.1 | 6.2 | 6.3 | 5.2 | 6.0 | 6.5 | 6.9 | 5.0 | 7.4 |
| 服务器总 CPU 使用率(%) | 28.7 | 34.8 | 40.9 | 47.0 | 53.0 | 59.3 | 65.6 | 71.3 | 77.1 |
| 额外开销占比(%) | 21.3 | 17.8 | 15.4 | 11.0 | 11.3 | 11.0 | 10.5 | 7.0 | 9.6 |
| 虚拟机数量 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

为了进一步验证服务器 CPU 额外开销保持稳定且与虚拟机数量无关,图 2 展示了虚拟机运行 Fhourstone 工作负载下,服务器 Steal Time 部分的 CPU 使用率,这个 CPU 使用率衡量的是虚拟机等待资源调度的平均等待时间.从图 2 可以发现:当 VCPU 数量超过 PCPU 数量时,出现明显的 CPU 额外开销,且这个额外开销与虚拟机数量的增长无关.

为了验证当 VCPU 数量小于 PCPU 数量时,服务器的 CPU 额外开销可忽略不计,我们使用了 LMbench 工作负载.图 3 展示虚拟机运行 Fhourstone 工作负载时的服务器 CPU 使用率及所有虚拟机的 CPU 使用率之和,虚拟

机数量由 1~3 增长.减去图 1 中的空负载情况下的服务器 CPU 使用率,我们发现:当虚拟机数量为 1 和 2 时,服务器的 CPU 额外开销几乎为 0;当虚拟机数量为 3 时,这个额外开销也仅为 2.1%.

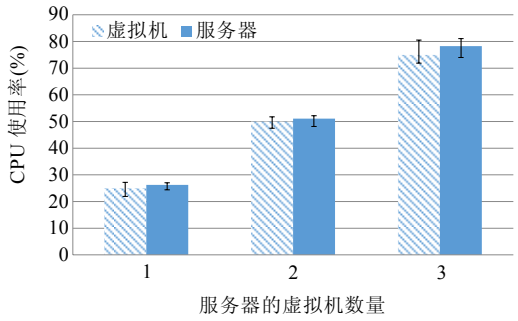


Fig.2 Server CPU resource consumption and additional virtual machine CPU resource consumption with LMbench workload

图 2 LMbench 工作负载下服务器 CPU 资源开销与虚拟机 CPU 资源开销

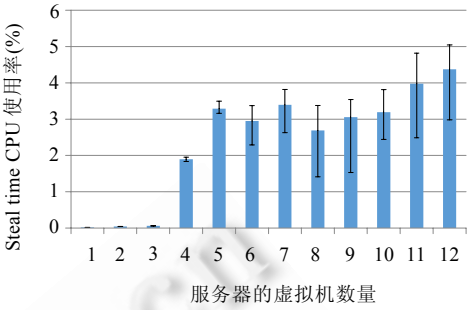


Fig.3 Server Steal time CPU resource consumption with Fhourstone workload

图 3 Fhourstone 工作负载下服务器 Steal time CPU 资源开销

## 5 资源预留整合算法仿真实验

为了验证资源预留整合(MRC)算法的性能,我们通过算法仿真实验比较 MRC 算法与常用的装箱问题算法 FFD 在服务器溢出概率方面的性能差异.根据第 3.4 节的描述,服务器溢出概率是保证云计算服务质量的重要参数,因此我们通过给定服务器数量,比较 MRC 和 FFD 的服务器溢出概率.本节首先介绍实验设置,然后分析实验结果.

### 5.1 实验设置

仿真实验运行于 Dell Optiplex 990,处理器为 4 核 Intel(R) Core(TM) i3-2120,主频为 3.30GHz,内存为 8GB,硬盘为 450GB SATA 硬盘.我们模拟了 1 000 个虚拟机的虚拟机整合环境,通过递增服务器数量,计算服务器 CPU 资源溢出概率.根据已有研究工作的结论[46],令虚拟机的 CPU 资源使用率满足正态分布.我们采用了 3 种正态分布模拟虚拟机 CPU 负载高中低 3 种情况,3 种正态分布的均值分别为 0.2,0.1,0.05,方差均为 0.01.对于资源调度等待开销所占的服务器 CPU 资源,我们采用第 4 节的实验测量结果,即 6.2%.值得注意的是:资源调度等待开销的 CPU 占有率与具体的虚拟机应用程序相关,但是并不影响 MRC 算法与 FFD 算法的性能差异.

### 5.2 实验结果

图 4~图 6 分别展示了高、中、低 3 种虚拟机 CPU 负载下,采用 FFD 算法以及 MRC 算法的服务器 CPU 资源溢出概率情况.

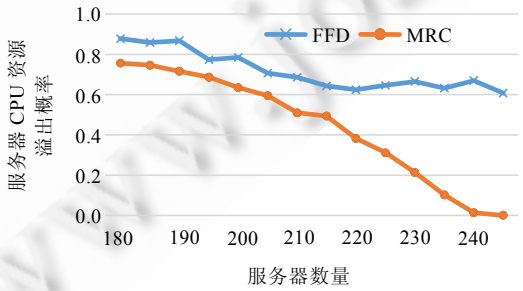


Fig.4 Server CPU resource overflow probability with high CPU load by FFD and MRC

图 4 高 CPU 负载下 FFD 算法与 MRC 算法的服务器 CPU 资源溢出概率

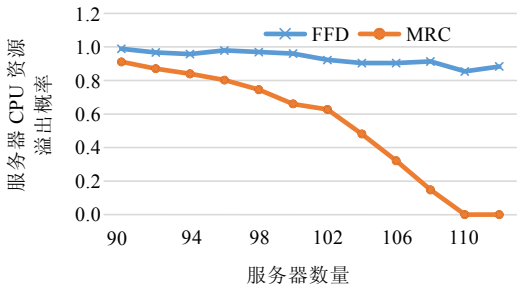


Fig.5 Server CPU resource overflow probability with medium CPU load by FFD and MRC

图 5 中 CPU 负载下 FFD 算法与 MRC 算法的服务器 CPU 资源溢出概率

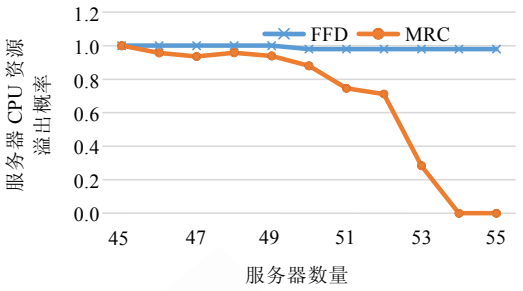


Fig.6 Server CPU resource overflow probability with low CPU load by FFD and MRC

图 6 低 CPU 负载下 FFD 算法与 MRC 算法的服务器 CPU 资源溢出概率

从图中可以看出:由于 MRC 算法为资源调度等待开销预留了一部分服务器 CPU 资源,相比于 FFD 算法,明显降低了服务器 CPU 资源溢出概率.

随着服务器数量的增加,MRC 算法的资源溢出概率也随之下降.然而,FFD 算法的资源溢出概率没有明显下降趋势,这是因为 FFD 算法没有考虑资源调度等待开销,低估了放置所有虚拟机所需的服务器数量,使得虚拟机没有得到足够的 CPU 资源.

## 6 总结与展望

本文使用 *M*/*M*/*n* 排队模型对 XCP 的 CPU 调度器进行建模分析,证明了在具有实际意义的约束条件下,虚拟机等待资源调度带来的服务器资源额外开销是存在的,且随着整合虚拟机数量的增长保持稳定.我们在考虑额外开销的基础上,提出资源预留整合算法(MRC)用于补充已有工作的不足.此外,我们在 XCP 上进行了典型工作负载的服务器资源开销测量实验,验证了理论分析结果.实验结果表明,这个额外开销平均占服务器 CPU 总开销的 11.7%.此外,我们在仿真实验中对比了 MRC 算法与常用的虚拟机整合算法 FFD,实验结果表明,MRC 算法相比于 FFD 算法,明显降低了服务器 CPU 资源溢出概率.

我们拟在以下 3 个方面进一步开展资源调度等待开销感知的虚拟机整合研究:① 基于计算密集型、内存密集型、存储密集型等不同类别的工作负载研究服务器 CPU 资源额外开销,并基于最大最小阈值预测方法确定服务器所需预留资源,改进资源预留整合算法;② 考虑在同一种工作负载下,异构服务器的 CPU 资源额外开销.在进行虚拟机整合过程中,通过给每一种架构的服务器赋予资源开销权重,模拟异构服务器的不同资源额外开销;③ 本文基于 XCP 的 CPU 调度器研究资源调度等待开销,而其他虚拟化产品使用不同的 CPU 调度机制,例如 KVM 使用 QEMU 模拟处理器,在 KVM 等其他虚拟化环境下的资源调度等待开销问题仍有待研究.

**References**:

[1] Kaplan JM, Forrest W, Kindler N. Revolutionizing data center energy efficiency. Technical Report, No.July-2008, McKinsey & Company, 2008.

[2] Koomey J. Growth in data center electricity use 2005 to 2010. Technical Report, No.August-1, Analytics Press, 2011.

[3] Scheihing P. DOE data center energy efficiency program. Technical Report, No.April-2009, U.S. Department of Energy, 2009.

[4] Birke R, Chen LY, Smirni E. Data centers in the wild: A large performance study. Technical Report, No.Z1204-002, IBM Research, 2012.

[5] VMware. 2013. http://www.vmware.com

[6] Xen. 2013. http://www.citrix.com/products/xenserver/overview.html

[7] VMware report: Server consolidation. 2013. http://www.vmware.com/consolidation/overview

[8] Bin packing problem. 2013. http://en.wikipedia.org/wiki/Bin_packing_problem

[9] Srikantaiah S, Kansal A, Zhao F. Energy aware consolidation for cloud computing. In: Proc. of the Conf. on Power Aware Computing and Systems (HotPower). Berkeley: USENIX Association, 2008. 10.

[10] Cardosa M, Korupolu MR, Singh A. Shares and utilities based power consolidation in virtualized server environments. In: Proc. of the 11th IFIP/IEEE Int'l Conf. on Symp. on Integrated Network Management (IM). 2009. 327−334. [doi: 10.1109/INM.2009.5188832]

[11] Feller E, Rilling L, Morin C. Energy-Aware ant colony based workload placement in clouds. In: Proc. of the 12th IEEE/ACM Int'l Conf. on Grid Computing. 2011. 26−33. [doi: 10.1109/Grid.2011.13]

[12] He LG, Zou DQ, Zhang Z, Jin H, Yang K, Jarvis SA. Optimizing resource consumptions in clouds. In: Proc. of the 12th IEEE/ACM Int'l Conf. on Grid Computing. 2011. 42−49. [doi: 10.1109/Grid.2011.15]

[13] Ghribi C, Hadji M, Zeghlache D. Energy efficient VM scheduling for cloud data centers: Exact allocation and migration algorithms. In: Proc. of the 13th IEEE/ACM Int'l Symp. on Cluster, Cloud and Grid Computing (CCGrid). 2013. 671−678. [doi: 10.1109/CCGrid.2013.89]

[14] Voorsluys W, Broberg J, Venugopal S, Buyya R. Cost of virtual machine live migration in clouds: A performance evaluation. In: Proc. of the 1st IEEE Int'l Conf. on Cloud Computing (CLOUD). 2009. 254−265. [doi: 10.1007/978-3-642-10665-1_23]

[15] Akoush S, Sohan R, Rice A, Moore AW, Hopper A. Predicting the performance of virtual machine migration. In: Proc. of the IEEE Int'l Symp. on Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS). 2010. 37−46. [doi: 10. 1109/MASCOTS.2010.13]

[16] Ye KJ, Jiang XH, Huang DW, Chen JH, Wang B. Live migration of multiple virtual machines with resource reservation in cloud computing environments. In: Proc. of the 3rd IEEE Int'l Conf. on Cloud Computing (CLOUD). 2011. 267−274. [doi: 10.1109/CLOUD.2011.69]

[17] Huang Q, Gao FQ, Wang R, Qi ZW. Power consumption of virtual machine live migration in clouds. In: Proc. of the 3rd Int'l Conf. on Communications and Mobile Computing (CMC). 2011. 122−125. [doi: 10.1109/CMC.2011.62]

[18] Strunk A, Dargie W. Does live migration of virtual machines cost energy. In: Proc. of the 27th IEEE Int'l Conf. on Advanced Information Networking and Applications (AINA). 2013. 514−521. [doi: 10.1109/AINA.2013.137s]

[19] Liu HK, Xu CZ, Jin H, Gong JY, Liao XF. Performance and energy modeling for live migration of virtual machines. In: Proc. of the 20th Int'l Symp. on High Performance Distributed Computing (HPDC). 2011. 171−182. [doi: 10.1145/1996130.1996154]

[20] Setzer T, Wolke A. Virtual machine re-assignment considering migration overhead. In: Proc. of the IEEE Network Operations and Management Symp. (NOMS). 2012. 631−634. [doi: 10.1109/NOMS.2012.6211973]

[21] Beloglazov A, Buyya R. Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers. In: Proc. of the 8th Int'l Workshop on Middleware for Grids, Clouds and e-Science (MGC). 2010. 1−6. [doi: 10.1145/1890799.1890803]

[22] 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. [doi: 10.1016/j.future.2011.04.017]

[23] Beloglazov A, Buyya R. Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints. IEEE Trans. on Parallel and Distributed Systems, 2013,24(7):1366−1379. [doi: 10.1109/TPDS.2012.240]

[24] Dong DP, Herbert J. Energy efficient VM placement supported by data analytic service. In: Proc. of the 13th IEEE/ACM Int'l Symp. on Cluster, Cloud and Grid Computing (CCGrid). 2013. 648−655. [doi: 10.1109/CCGrid.2013.94]

[25] Berral JL, Gavalda R, Torres J. Adaptive scheduling on power-aware managed data-centers using machine learning. In: Proc. of the 12th IEEE/ACM Int'l Conf. on Grid Computing. 2011. 66−73. [doi: 10.1109/Grid.2011.18]

[26] Wang LX, Xu J, Zhao M. Application-Aware cross-layer virtual machine resource management. In: Proc. of the 9th Int'l Conf. on Autonomic Computing (ICAC). 2012. 13−22. [doi: 10.1145/2371536.2371541]

[27] Feller E, Morin C, Esnault A. A case for fully decentralized dynamic VM consolidation in clouds. In: Proc. of the 4th IEEE Int'l Conf. on Cloud Computing Technology and Science (CloudCom). 2012. 26−33. [doi: 10.1109/CloudCom.2012.6427585]

[28] Maguluri ST, Srikant R, Ying L. Stochastic models of load balancing and scheduling in cloud computing clusters. In: Proc. of the IEEE INFOCOM. 2012. 702−710. [doi: 10.1109/INFCOM.2012.6195815]

[29] Xiao Z, Song WJ, Chen Q. Dynamic resource allocation using virtual machines for cloud computing environment. IEEE Trans. on Parallel and Distributed Systems, 2013,24(6):1107−1117. [doi: 10.1109/TPDS.2012.283]

[30] Xu H, Li BC. Anchor: A versatile and efficient framework for resource management in the cloud. IEEE Trans. on Parallel and Distributed Systems, 2013,24(6):1066−1076. [doi: 10.1109/TPDS.2012.308]

[31] Zhang Q, Zhani MF, Zhang S, Zhu QY, Boutaba R, Hellerstein JL. Dynamic energy-aware capacity provisioning for cloud computing environments. In: Proc. of the 9th Int'l Conf. on Autonomic Computing (ICAC). 2012. 145−154. [doi: 10.1145/2371536.2371562]

[32] Shi L, Butler B, Botvich D, Jennings B. Provisioning of requests for virtual machine sets with placement constraints in IaaS clouds. In: Proc. of the 15th IFIP/IEEE Int'l Symp. on Integrated Network Management (IM). IEEE, 2013. 499−505.

[33] Zheng ZY, Li MM, Xiao X, Wang JP. Coordinated resource provisioning and maintenance scheduling in cloud data centers. In: Proc. of the IEEE INFOCOM. 2013. 345−349. [doi: 10.1109/INFCOM.2013.6566792]

[34] Kliazovich D, Bouvry P, Khan SU. DENS: Data center energy-efficient network-aware scheduling. In: Prof. of the IEEE/ACM Int'l Conf. on Green Computing and Communications & Int'l Conf. on Cyber, Physical and Social Computing (GREENCOMCPSCOM). 2010. 69−75. [doi: 10.1007/s10586-011-0177-4]

[35] Sonnek J, Greensky J, Reutiman R, Chandra A. Starling: Minimizing communication overhead in virtualized computing platforms using decentralized affinity-aware migration. In: Proc. of the 39th Int'l Conf. on Parallel Processing (ICPP). 2010. 228−237. [doi: 10.1109/ICPP.2010.30]

[36] Wang M, Meng XQ, Zhang L. Consolidating virtual machines with dynamic bandwidth demand in data centers. In: Proc. of the IEEE INFOCOM. 2011. 71−75. [doi: 10.1109/INFCOM.2011.5935254]

[37] Breitgand D, Epstein A. Improving consolidation of virtual machines with risk-aware bandwidth oversubscription in compute clouds. In: Proc. of the IEEE INFOCOM. 2012. 2861−2865. [doi: 10.1109/TPDS.2012.241]

[38] Biran O, Corradi A, Fanelli M, Nus A, Raz D, Silvera E. A stable network-aware VM placement for cloud systems. In: Proc. of the 12th IEEE/ACM Int'l Symp. on Cluster, Cloud and Grid Computing (CCGrid). 2012. 498−506. [doi: 10.1109/CCGrid.2012.119]

[39] Dong JK, Jin X, Wang HB, Li YY, Zhang P, Cheng SD. Energy-Saving virtual machine placement in cloud data centers. In: Proc. of the 13th IEEE/ACM Int'l Symp. on Cluster, Cloud and Grid Computing (CCGrid). 2013. 618−624. [doi: 10.1109/CCGrid.2013.107]

[40] Ghorbani S, Caesar M. Walk the line: Consistent network updates with bandwidth guarantees. In: Proc. of the 1st Workshop on Hot Topics in Software Defined Networks. 2012. 67−72. [doi: 10.1145/2342441.2342455]

[41] Mann V, Gupta A, Dutta P, Vishnoi A, Bhattacharya P, Poddar R, Iyer A. Remedy: Network-Aware steady state VM management for data centers. In: Proc. of the 11th Int'l IFIP TC 6 Conf. on Networking. 2012. 190−204. [doi: 10.1007/978-3-642-30045-5_15]

[42] Wang XD, Yao YJ, Wang XR, Lu KF, Cao Q. CARPO: Correlation-Aware power optimization in data center networks. In: Proc. of the IEEE INFOCOM. Orlando: IEEE, 2012. 1125−1133.

[43] Cohen R, Lewin-Eytan L, Naor J, Raz D. Almost optimal virtual machine placement for traffic intense data centers. In: Proc. of the IEEE INFOCOM. 2013. 355−359. [doi: 10.1109/INFCOM.2013.6566794]

[44] Popa L, Kumar G, Chowdhury M, Krishnamurthy A, Ratnasamy S, Stoica I. FairCloud: Sharing the network in cloud computing. In: Proc. of the ACM SIGCOMM. 2012. 187−198. [doi: 10.1145/2070562.2070584]

[45] Meng XQ, Pappas V, Zhang L. Improving the scalability of data center networks with traffic-aware virtual machine placement. In: Proc. of the IEEE INFOCOM. San Diego: IEEE, 2010. 1−9.

[46] Shrivastava V, Zerfos P, Lee KW, Jamjoom H, Liu YH, Banerjee S. Application-Aware virtual machine migration in data centers. In: Proc. of the IEEE INFOCOM. 2011. 66−70. [doi: 10.1109/INFCOM.2011.5935247]

[47] Dias DS, Costa LHMK. Online traffic-aware virtual machine placement in data center networks. In: Proc. of the Global Information Infrastructure and Networking Symp. (GIIS). 2012. 1−8. [doi: 10.1109/GIIS.2012.6466665]

[48] Zhang BL, Qian ZZ, Huang W, Li X, Lu SL. Minimizing communication traffic in data centers with power-aware VM placement. In: Proc. of the 6th Int'l Conf. on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS). 2012. 280−285. [doi: 10.1109/IMIS.2012.71]

[49] Hu LT, Schwan K, Gulati A, Zhang JJ, Wang CW. Net-Cohort: Detecting and managing VM ensembles in virtualized data centers. In: Proc. of the 9th Int'l Conf. on Autonomic Computing (ICAC). 2012. 3−12. [doi:10.1145/2371536.2371540]

[50] Stage A, Setzer T. Network-Aware migration control and scheduling of differentiated virtual machine workloads. In: Proc. of the ICSE Workshop on Software Engineering Challenges of Cloud Computing. 2009. 9−14. [doi: 10.1109/CLOUD.2009.5071527]

[51] Alicherry M, Lakshman TV. Network aware resource allocation in distributed clouds. In: Proc. of the IEEE INFOCOM. Orlando: IEEE, 2012. 963−971.

[52] Jain N, Menache I, Naor J, Shepherd FB. Topology-Aware VM migration in bandwidth oversubscribed datacenter networks. In: Proc. of the 39th Int'l Colloquium Conf. on Automata, Languages, and Programming. 2012. 586−597. [doi: 10.1007/978-3-642-31585-5_52]

[53] Giurgiu I, Castillo C, Tantawi A, Steinder M. Enabling efficient placement of virtual infrastructures in the cloud. In: Proc. of the 13th Int'l Middleware Conf. 2012. 332−353. [doi: 10.1007/978-3-642-35170-9_17]

[54] Jiang JW, Lan T, Ha S, Chen MH, Chiang M. Joint VM placement and routing for data center traffic engineering. In: Proc. of the IEEE INFOCOM. 2012. 2876−2880. [doi: 10.1109/INFCOM.2012.6195719]

[55] Gao Y, Stolyar AL, Walid A. Shadow-Routing based dynamic algorithms for virtual machine placement in a network cloud. In: Proc. of the IEEE INFOCOM. Turin: IEEE, 2013. 620−628.

[56] Yang JS, Liu PF, Wu JJ. Workload characteristics-aware virtual machine consolidation algorithms. In: Proc. of the 4th IEEE Int'l Conf. on Cloud Computing Technology and Science (CloudCom). 2012. 42−49. [doi: 10.1109/CloudCom.2012.6427540]

[57] Lu T, Chen MH, Andrew LLH. Simple and effective dynamic provisioning for power-proportional data centers. IEEE Trans. on Parallel and Distributed Systems, 2013,24(6):1161−1171. [doi: 10.1109/TPDS.2012.241]

[58] Palanisamy B, Singh A, Liu L, Jain B. Purlieus: Locality-Aware resource allocation for MapReduce in a cloud. In: Proc. of the 2011 Int'l Conf. for High Performance Computing, Networking, Storage and Analysis. 2011. 1−11. [doi: 10.1145/2063384.2063462]

[59] Huang Z, Tsang DHK, She J. A virtual machine consolidation framework for MapReduce enabled computing clouds. In: Proc. of the 24th Int'l Teletraffic Congress. Krakow: IEEE, 2012. 1−8.

[60] Hwang E, Kim KH. Minimizing cost of virtual machines for deadline-constrained MapReduce applications in the cloud. In: Proc. of the 13th ACM/IEEE Int'l Conf. on Grid Computing. 2012. 130−138. [doi: 10.1109/Grid.2012.19]

[61] Li M, Subhraveti D, Butt AR, Khasymski A, Sarkar P. CAM: A topology aware minimum cost flow based resource manager for MapReduce applications in the cloud. In: Proc. of the 21st Int'l Symp. on High-Performance Parallel and Distributed Computing. 2012. 211−222. [doi: 10.1145/2287076.2287110]

[62] Alicherry M, Lakshman TV. Optimizing data access latencies in cloud systems by intelligent virtual machine placement. In: Proc. of the IEEE INFOCOM. 2013. 647−655. [doi: 10.1109/INFCOM.2013.6566850]

[63] Rodero I, Jaramillo J, Quiroz A, Parashar M, Guim F. Towards energy-aware autonomic provisioning for virtualized environments. In: Proc. of the 19th ACM Int'l Symp. on High Performance Distributed Computing (HPDC). 2010. 320−323. [doi: 10.1145/1851476.1851520]

[64] Credit scheduler. 2013. http://wiki.xen.org/wiki/CreditScheduler

[65] Cherkasova L, Gupta D, Vahdat A. Comparison of the Three CPU Schedulers in Xen. Xen Summit Spring, 2007.

[66] Chisnall D. The Definitive Guide to the Xen Hypervisor. Persion Education, 2007.

[67] Takács L. Introduction to the Theory of Queues. New York: Oxford University Press, 1962.

[68] Fhourstone. 2013. http://homepages.cwi.nl/~tromp/c4/fhour.html

[69] cpulimit. 2013. http://cpulimit.sourceforge.net

[70] LMbench. 2013. http://www.bitmover.com/lmbench

**李铭夫**(1987－),男,广东珠海人,博士生,CCF 学生会员,主要研究领域为云计算资源调度.
E-mail: limingfu@ict.ac.cn

**李忠诚**(1962－),男,博士,研究员,博士生导师,CCF 高级会员,主要研究领域为计算机网络.
E-mail: zcli@ict.ac.cn

**毕经平**(1974－),女,博士,研究员,博士生导师,CCF 高级会员,主要研究领域为下一代互联网,网络性能测试.
E-mail: jpingbi@ict.ac.cn