文章编号：0253-374X(2014)05-0782-08 DOI:10.3969/j.issn.0253-374x.2014.05.020

一种基于时分复用的云资源管理方法

匡桂娟[1,2,3]，曾国荪[1,3]

(1. 同济大学 电子与信息工程学院，上海 201804；2. 青岛农业大学 理学与信息学院，山东 青岛 266109；
3. 国家高性能计算机工程技术中心同济分中心，上海 201804)

摘要：为了实现多租户的云计算环境下用户对计算资源的公平使用，用有限资源应对最大量的用户需求，针对云计算中成批到达的可划分独立任务，研究了一种基于时分复用的虚拟化资源管理方法. 将整个云中计算资源作为复用对象，根据用户任务需求确定资源的复用周期和时间片，给出了资源时分复用的多种策略，并且对不同策略进行了深入的性能指标分析，形成了相应结论，以应对和指导不同应用场景. 最后验证了资源管理方法的有效性.

关键词：云计算；批任务调度；资源虚拟化；时分复用
中图分类号：TP338 **文献标志码**：A

Time-division Multiplexing-based Cloud Resource Management Methods

KUANG Guijuan [1,2,3], *ZENG Guosun*[1,3]

(1. College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China; 2. School of Science and Information Science, Qingdao Agricultural University, Qingdao 266109, China; 3. Tongji Branch, National Engineering & Technology Center of High Performance Computer, Shanghai 201804, China)

Abstract: Focusing on the dividable batch tasks running in the cloud computing environment, a study is made of the resource virtualization based on time-division multiplexing in order to ensure a fair opportunity for users in cloud computing with multi-tenants and satisfy the requirements as much as possible. Computational resource in cloud system is viewed as the time-multiplexing object. According to users' requirements, the appropriate multiplexing cycle and time slices are determined with the proposed strategies. Furthermore, a detailed performance analysis for different strategies is made. Theoretical analysis and simulations prove the time-division multiplexing resource management method to be valid.

Key words: cloud computing; batch tasks scheduling; resource virtualization; time-division multiplexing

云计算是在并行计算、分布式计算、集群计算、网格计算等基础上发展起来的一种基于网络的计算模式，以服务的形式对外提供 IT 资源，包括基础设施服务 IaaS(infrastructure as a service)、平台服务 PaaS(platform as a service)、软件服务 SaaS(software as a service)等，并且具有 SLA(service level agreement)保障[1-2]. 对数据中心而言，有效整合云数据中心的资源，提高资源利用率，节约能源，降低运行成本是云数据中心关注的热点[3]. 对于用户而言，可以通过 Internet 随时随地享有公平获取服务的机会. 实现云计算资源管理自动部署，动态扩展，按需分配，方便用户按需获取资源是云计算研究的热点和难点. 2010 年 Zenoss 的调查报告指出云资源管理是仅次于云安全问题的一大挑战[4].

云资源管理的困难不仅在于云中心要管理的资源规模庞大，还来自于它要同时为多租户提供服务这一云的本质特征[5]. 师雪霖等[6]指出，目前云计算中许多资源管理方法仍沿用传统的网格资源的管理方法，例如采用线性规划的数学模型，运用启发式算法对问题求解等[7]. 这些算法没有很好地考虑到云计算的应用特点，大多数仍然只关注任务的最短完成时间. 已有的较好的云资源管理系统多从云运营商角度出发进行管理，追求的是资源的整合和负载的均衡，如基于 CPU 利用率的资源管理[8-9]，基于能耗的资源管理[10]等，也没有很好地关注云计算环境

收稿日期：2013-08-27
基金项目：国家“八六三”高技术研究发展计划(2009AA012201)；国家自然科学基金(61272107,61202173,61103068)；上海市优秀学科带头人计划(10XD1404400)；教育部高等学校博士学科点专项科研基金(20090072110035)；教育部网络时代的科技论文快速共享专项研究(20110740001)；华为创新计划(IRP-2013-12-03)
第一作者：匡桂娟(1972—)，女，讲师，博士生，主要研究方向为并行分布处理及云计算. E-mail:1gjkuang@tongji.edu.cn
通讯作者：曾国荪(1964—)，男，教授，博士生导师，工学博士，主要研究方向为并行计算、可信软件及信息安全.
E-mail:gszeng@tongji.edu.cn

下用户要求公平使用资源的愿望. 云数据中心资源管理的关键技术是虚拟化技术,目前运用较多的虚拟化技术有 Xen[11],KVM[12] 以及 Vmware[13] 等,主要是以虚拟机的形式向不同用户提供资源,例如 Amazon EC2[14]. 以上这些虚拟化技术属于"以大变小"的应用模式,即将一台功能强大的物理计算资源虚拟成多个独立的、满足不同要求的小计算资源,服务不同的用户,在一定程度上提高了物理资源的利用率. 虚拟化技术另外一种应用模式是"由小组大",即将多个孤立的、小的物理资源聚合成一个更强大的服务器,完成特定的功能. 例如 Linux 的开源项目 LVS(linux virtual server)[15]. 目前虚拟化技术虽然较好地实现了多用户对计算资源的共享和动态分配,一定程度上提高了资源利用率,但仍然存在很多问题:计算资源浪费现象仍大量存在,因为虚拟机管理机制中按用户峰值要求,即最高需求量分配资源,通常情况下用户未必需要如此多的资源;虚拟机之间资源不能共享,导致虚拟机资源利用不足;另外,云计算中尤为突出的用户公平使用资源的要求难以有效保证;有限资源如何应对几乎无限的用户请求问题尚未引起足够重视. 因此,现有虚拟化管理技术还远不能满足按需服务的要求,本文针对云计算中成批到达的可划分的独立任务,探索一种基于时分复用的虚拟化资源管理方法.

事实上,在网络并行计算系统中,存在着大量独立的可切分应用程序. 例如蒙特卡罗(Monte Carlo)模拟、分形计算(如 Mandelbrot)、海量信息检索(如 key breaking)、参数扫描、图像处理(如层析成像重建)、数据挖掘等都可看成是独立可切分计算任务. 最近流行的 MapReduce 并行编程模型,特别适合独立的可切分任务的编程求解. 本文正是针对批到达的独立可划分任务运行在云计算环境中,给出一种基于时分复用的资源管理方法,旨在公平有效地使用云资源,并有效地提高系统资源的利用率.

1　云资源虚拟化的一般过程

1.1　云资源按需分配的要求

云资源分配的过程就是安排计算资源来应对用户提出的各种需求,达到期待的目标. 该目标对于用户来讲首先是计算时间少,任务的等待时间小,分配到的计算资源利用率高,可靠性高等方面;对于资源提供方来讲,因为云计算是规模经济效益驱动下的计算模式,因此云平台一方面要尽量提高资源利用率,另外一方面更希望同时将资源提供给更多的用户,并且给用户公平的资源使用体验. 由于云资源多样性,以及用户需求的多样性,满足该目标并不是一件容易的事. 图 1 给出了云计算资源按需分配的示意图.

在图 1 中,最上层,用户向云系统提交应用任务,并提出资源请求. 用户提交给云系统的应用任务多种多样,可能任务类型不同,例如科学计算、事务处理、实时控制等;可能是通用计算,也可能是专用计算;可能任务可以划分实现并行执行,也可能任务不可划分只能串行执行;可能用户的 QoS(Quality of Service)不尽相同,等等. 最底层,资源提供商向云系统提供物理计算资源. 物理资源可能是异构的、分布的、动态的、自治的. 例如计算节点可能是 PC 机、工作站、小型机、超级计算机,也可能是 SIMD (Single Instruction Multiple Data)、MIMD (Multiple Instruction Multiple Data)、向量机、专用机等. 中间层,管理模块根据用户的 QoS 需求提供与之匹配的物理资源,即将物理资源映射为满足用户需求的逻辑资源.

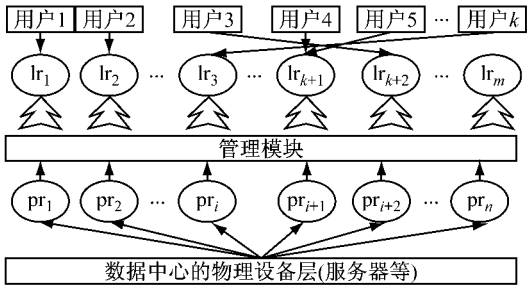


图 1　云计算按需分配资源示意图

Fig. 1　Overview of on-demand resource management in cloud computing system

正如上面所述,在图 1 中,物理计算资源 pr_1, pr_2,…, pr_m 很可能无法应对大量和多样的用户请求 ur_1, ur_2,…, ur_n,需要有效的虚拟技术对物理资源进行组织管理,形成逻辑资源 lr_1, lr_2,…, lr_m,以便满足用户需求. 这种把有限固定物理资源按需重新规划构成逻辑资源,以便提高系统利用率的管理技术就是虚拟化技术. 本文研究的时分复用技术属于虚拟化技术的一种.

1.2　资源分配的基本概念

定义 1　物理资源 (physical resource):是指在云计算环境中,直接参与计算处理的物理器件,又称计算节点,可用一个五元组表示 $pr=(\lambda, \tau, v, c, f)$,整个云系统的物理资源记为 $PRS=\{pr_1, pr_2,$

…, pr_n}.

在上面定义中:λ 是指物理资源的计算机类型,例如串行计算机、并行计算机、向量计算机、信号处理专用机等;τ 是指物理资源的时钟中断周期,是物理资源处理动作的基本时间单位;v 是计算节点的处理速度,用来表征节点计算能力的大小;c 是指计算节点进行任务切换的时间开销;f 是计算节点特点,$f\in\{0,1,2\}$,"0"代表该计算节点不可再划分,即原子器件,"1"表示计算节点空间上可划分,"2"表示计算节点时间上可划分.

定义 2 用户需求 (user requirement):是指请求云平台提供计算资源、执行任务处理、满足服务质量的要求,即请求任务,单个用户需求可用一个四元组表示 $ur=(\lambda, w, g, \tau)$. 整个云系统的用户需求记为 $URS=\{ur_1, ur_2, \cdots, ur_m\}$.

在上面定义中:λ 是指用户请求任务的业务类型,例如科学计算、事务处理、实时控制等;w 是用户请求任务的计算工作量大小;g 是指任务可切分的粒度,记 $g=1$ 表示用户任务可以任意切分,显然 $g=w$ 表示任务不可切分;τ 是用户任务要求的最晚完成时间,即截止时间(deadline).

定义 3 逻辑资源(logical resource):又称虚拟资源,是指利用虚拟化技术对物理资源进行划分、组合、复用等操作,以便满足用户需求,重新整合的计算资源,是一种用户视角的计算资源. 单个逻辑资源可用一个三元组表示 $lr=(\lambda, v, SP)$, 整个云系统的物理资源记为 $LRS=\{lr_1, lr_2, \cdots, lr_n\}$.

在上面定义中:λ 是指逻辑资源虚拟化后的计算类型;v 是逻辑资源的综合处理速度,表征逻辑资源的计算能力;$SP=(sp_1, sp_2, \cdots, sp_i \cdots, sp_k)$, k 表示组成逻辑资源 lr 用到的物理计算资源种类数,sp_i 表示组成逻辑资源的第 i 种物理资源的类型和数量.

2 资源时分复用的方法

2.1 计算资源时分复用的基本思想

对于每一个计算资源,将其整个运行时间划分成一个个小的时间片段,让使用该资源的多个用户任务轮流占用这些时间片段. 其中一个用户任务一次只占用一个时间片段,当所有用户任务都被运行一个时间片后,该计算资源的一个运行周期结束,进入下一个运行周期. 如此周而复始,直到所有的用户请求任务被执行完成. 资源时分复用的优点是:① 每个用户得到公平使用资源的权利,符合云计算中多租户的特点;② 有限的物理计算资源可以应对尽可能多的用户需求;③ 单个任务不独占物理资源,多个用户通过小的时间片共享一个物理资源,提高了资源的利用率;④ 用户只需要在自己分配到的时间片内执行任务即可,而不需要关心底层如何实现的,提高了资源使用的透明度. 时分复用的工作原理如图 2 所示.

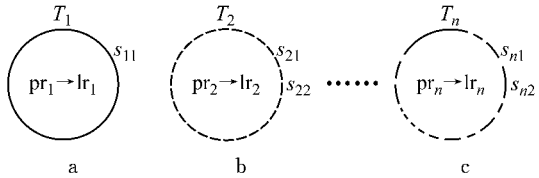


图 2 物理资源的时分复用原理

Fig.2 Time-division multiplexing principle

图 2a 中,资源 pr_1 的运行周期 T_1 是一个没有被切分的周期,表示资源被任务独占的执行模式. 图 2b 中,资源 pr_2 的运行周期 T_2 被均等划分,表示资源分配给每个任务相等的时间片的执行模式. 图 2c 中,资源 pr_n 的周期 T_n 被不均等划分,表示资源分配给每个任务不等的时间片的执行模式. 实施复用之后的资源称为逻辑资源.

2.2 资源时分复用的模型

对于云计算环境中的每个计算节点,实施复用之后具有运行周期 T、时间片 s_{ij} 等概念和变量. T 是每个资源为所有用户任务服务一轮花费的时间,又称资源的复用周期. 将 T 分成多个小的时间片即 s_{ij},表示第 i 个计算节点为第 j 个用户任务分配的时间段. s_{ij} 的大小可以相同,也可以不同. 每一个物理资源都可以独立地形成相应的逻辑资源,每个时间片都可以被用来处理任意用户请求任务. 要合理有效地划分计算节点的运行时间,必须考虑多方面因素,包括一个计算节点运行周期 T 的大小,时间片 s_{ij} 的大小,一个运行周期内时间片的数目,以及用户请求任务与时间片的对应关系等.

定义 4 复用后的物理资源:是一种特殊的虚拟资源,即采用复用的虚拟化技术,使得物理资源虚拟化后,逻辑上可以重复多次被多个用户任务同时使用的资源,可用一个四元组表示为 $lr'=(\lambda, v, T, S)$. 其中,λ 是该资源的计算类型,v 是该资源的处理速度,T 是该资源复用后的运行周期,$S=\{s_1, s_2, \cdots, s_i \cdots, s_m\}$ 为复用周期中划分的时间片,s_i 是第 i 个时间片,m 是时间片的总个数. 物理资源复用后的逻辑资源的集合为 $LRS'=\{lr'_1, lr'_2, \cdots, lr'_n\}$.

将复用后的物理资源集合 LRS′中不同 lr′的不

同时间片 s_{ij} 进行组合,构成用户可以使用的逻辑资源 LRS=$\{\mathrm{lr}_1, \mathrm{lr}_2, \cdots, \mathrm{lr}_m\}$.

根据上述分析,沿用上述符号,结合前面定义 3 用户视角的虚拟资源定义,云资源复用问题的形式化描述如下:

$$\begin{cases} \exists\, T_i, s_{ij}: \{\mathrm{pr}_1, \mathrm{pr}_2, \cdots, \mathrm{pr}_n\} \rightarrow \{\mathrm{lr}'_1, \mathrm{lr}'_2, \cdots, \mathrm{lr}'_n\} \\ \quad \wedge \{\mathrm{lr}'_1, \mathrm{lr}'_2, \cdots, \mathrm{lr}'_n\} \rightarrow \{\mathrm{lr}_1, \mathrm{lr}_2, \cdots, \mathrm{lr}_m\} \\ \mathrm{lr}_j = \bigcup\limits_{i=1,2,\cdots,n} \{\mathrm{lr}'_i . s_{ij}\} \\ \mathrm{ur}_j . \tau = \sum\limits_i s_{ij} \\ T_i = \sum\limits_{j=1}^{m} s_{ij} + m \cdot \mathrm{lr}'_i . c \\ \mathrm{ur}_j . \tau \leqslant \theta_j,\ \theta_j \text{ 是 } \mathrm{ur}_j \text{ 的截止完成时间} \\ \quad 1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m \end{cases}$$

2.3　资源复用评价指标

定义 5　任务组完成时间(makespan):指一组用户请求向系统最早提交任务时刻开始,到所有任务都执行完成为止,系统运行的一个时间跨度,记为 $t_{\mathrm{ms}} = \max\limits_{1\leqslant i\leqslant m} t_{\mathrm{fi}} - \min\limits_{1\leqslant i\leqslant m} t_{\mathrm{ai}}$,其中 t_{fi} 表示用户请求任务 ur_i 的完成时刻,t_{ai} 表示用户请求任务 ur_i 到达系统的时刻.

定义 6　任务组平均响应时间(average response time):指一组用户请求的第一个任务进入系统到所有任务都执行为止,每个任务保持在系统中的平均时间,记为 $t_{\mathrm{ar}} = \dfrac{1}{m}\sum\limits_{i=1}^{m}(t_{\mathrm{fi}} - t_{\mathrm{ai}})$.

定义 7　任务组平均延迟率(average slowdown rate):指用户请求任务的响应时间与系统执行该任务的运行时间的平均比值,记为 $r_{\mathrm{as}} = \dfrac{1}{m}\sum\limits_{i=1}^{m}\left(\dfrac{t_{\mathrm{fi}} - t_{\mathrm{ai}}}{t_{\mathrm{ei}}}\right)$,其中 t_{ei} 是计算资源运行用户请求任务 ur_i 的实际时间.

定义 8　资源利用率(resource utilization):指资源用于用户请求任务的有效工作能力与资源总的工作能力的比值,记为 $r_{\mathrm{u}} = \dfrac{(\sum\limits_{i=1}^{m} t_{\mathrm{ei}}) \cdot \mathrm{pr}_{\mathrm{ur}_i} . c}{(\sum\limits_{i=1}^{n} \mathrm{pr}_i . c) \cdot t_{\mathrm{ms}}}$. 其中 $\mathrm{pr}_{\mathrm{ur}_i} . c$ 代表用户请求 ur 被分配到的物理资源 $\mathrm{pr}_{\mathrm{ur}}$ 的切换开销.

定义 9　系统任务服务率(service rate):指系统按照用户要求完成的用户请求任务个数 l 与提交给系统总的用户请求任务个数 m 的比值,记为 $r_{\mathrm{s}} = l/m, l<m$.

3　执行批任务的资源时分复用策略

3.1　条件和前提假设

不妨假设云计算环境中用户请求任务成批到达,记单批任务到达时刻为 0,即每个用户请求到达都在同一个时刻,但用户任务要求的完成时间可能不同,任务的负载量可能不同,可划分的粒度可能不同. 这样的假设具有一定的合理性和实效性,虽然用户提交服务请求在时间上可能不是严格同时成批到达的,但是目前云计算应用服务多是大数据事务处理任务,实时控制服务任务不是太多,因此可以将时间上相近到达的多个任务,看做和处理成时间上同时到达的批任务. 用户的要求可以通过任务的截止时间来反映.

假设用户请求任务是独立的可任意切分的. 例如图像处理、生物信息学、粒子物理、气象模拟等领域存在大量可切分的计算任务. 对于关联和不可切分计算任务,将在后续工作中进一步研究.

假设成批到达的任务个数为 m,云计算环境中物理资源的个数为 n. 假设所有物理资源都是可时分复用的计算资源,所有用户请求任务都可在所有物理资源上运行,所有任务执行完之前所有资源一直处于运行状态.

假设物理资源复用切换开销都相等,如此假设目的是为了分析讨论方便,不会影响方法的有效性. 用 $c_0 = \max\limits_{1\leqslant j\leqslant n} \mathrm{pr}_j . c$,表示所有资源的任务切换开销,那么 m 次任务切换开销用 c 表示,$c = mc_0$.

3.2　优先和独占策略

策略 1　最小任务优先调度到最快资源上执行:该策略的思想是每次选择一个任务将其分配到某个资源上,并独占该资源直到任务执行完成,如图 2a 所示,选择时要满足以下条件:① 被选择的任务工作量最小;② 被选择的资源执行速度最快;此时任务在该资源上的预计完成时间是最短的.

策略 2　最大任务优先调度到最快资源上执行:该策略的思想是每次选择一个任务将其分配到某个资源上,并独占该资源直到任务执行完成,如图 2a 所示,选择时要满足以下条件:① 被选择的任务工作量最大;② 被选择的资源执行速度最快. 该策略有利于系统负载均衡.

假设按照策略 1 或者策略 2,资源 pr_j 上运行了 x_j 个用户请求任务,分别用 $\mathrm{ur}_{j1}, \mathrm{ur}_{j2}, \cdots, \mathrm{ur}_{jx_j}$ 表示. 同时记 $t_{\mathrm{ms}}^{(1)}$ 表示策略 1 中的 t_{ms} 值,后文中其他策略

的性能评价指标的表示与此类同,则根据2.3节中评价指标的定义,优先和独占策略具有以下性能指标:

(1) 任务组的完成时间

$$t_{\mathrm{ms}}{}^{(1)} = t_{\mathrm{ms}}{}^{(2)} = \max_{1\leqslant j\leqslant n}\left(\sum_{y=1}^{x_j}\frac{\mathrm{ur}_{jy}.w}{\mathrm{pr}_j.v}\right) \tag{1}$$

(2) 任务组的平均响应时间

$$t_{\mathrm{ar}}^{(1)} = t_{\mathrm{ar}}^{(2)} = \frac{1}{m}\sum_{j=1}^{n}\sum_{y=1}^{x_j}\left(\sum_{z=1}^{y}\frac{\mathrm{ur}_{j_z}.w}{\mathrm{pr}_j.v}\right) \tag{2}$$

其中,内层求和表示机器 pr_j 上第 y 个任务的响应时间;次外层求和表示该机器上所有任务响应时间的和;最外层和是所有任务响应时间的和.

(3) 任务组平均延迟率

$$r_{\mathrm{as}}{}^{(1)} = r_{\mathrm{as}}{}^{(2)} = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{t_{fi}-t_{ei}}{t_{ei}}\right) \tag{3}$$

(4) 资源利用率为

$$r_{\mathrm{u}}^{(1)} = r_{\mathrm{u}}^{(2)} = \frac{\left(\sum_{i=1}^{m}t_{ei}\right)\cdot \mathrm{pr}_{\mathrm{ur}_i}.c}{\left(\sum_{i=1}^{n}\mathrm{pr}_j.c\right)\cdot t_{\mathrm{ms}}} \tag{4}$$

其中,$\mathrm{pr}_{\mathrm{ur}_i}\cdot c$ 代表用户请求被分配到的物理资源.

3.3 时分复用策略

策略3 相等时间片相等任务量的时分复用策略:该策略的主要思想是:如图2b所示,云系统中每个复用的物理资源,即逻辑资源,为每一个用户任务分配一个固定和相等的时间片,每一个时间片中不同计算节点处理的任务量也是相等的,以便云系统公平地执行每个用户任务.当所有任务都执行了一个时间片以后,就说该计算节点运行了它的一个时分复用周期.如此轮转和周期地执行下去,直到所有的任务都完成.可见,每个用户都能够感觉到均等地享有计算资源的机会.

本策略中固定和相等的时间片用 s_α 表示,合理设定 s_α 要求考虑以下几个因素:

(1) 计算节点的时钟中断周期,应满足 $s_\alpha = k\cdot \mathrm{lcm}(\{\mathrm{pr}_j.\tau \mid j=1,2,\cdots,n\})$,其中 k 是待定的整数值.因为时钟中断周期是计算节点的最小处理时间单位,为了使每个计算节点都能够在一个时间片内至少执行一个最基本的处理操作,s_α 应该大于等于所有时钟中断周期的最小公倍数.但时间片过小会引起任务频繁切换,导致系统开销大.

(2) 应满足 $\frac{s_\alpha}{c_0+s_\alpha}\geqslant\rho$,其中 $c_0=\max\limits_{1\leqslant j\leqslant n}\mathrm{pr}_j.c$,即将所有计算节点的任务切换开销看成是相同的,均为 c_0.公式中的 ρ 是对系统一个时间片中任务运行时间所占比例的一个要求.

(3) 考虑任务量与资源处理速度因素,应满足 $s_\alpha \leqslant \frac{1}{n}\cdot\frac{\min\limits_{1\leqslant i\leqslant m}\mathrm{ur}_i.w}{\max\limits_{1\leqslant j\leqslant n}\mathrm{pr}_j.v}$,不等式的右边是最小任务在最大能力计算资源上的运行时间.

本策略中一个时间片内处理的固定和相等的任务量用 w_α 表示,$w_\alpha=\min\limits_{1\leqslant j\leqslant n}(\mathrm{pr}_j.v\cdot s_\alpha)$,以保证每一个逻辑资源在一个时间片内,拥有充足的任务工作量,为每个任务都执行相等的工作量.

根据时分复用的原理,如图2b所示,策略3具有以下性能指标:

(1) 每个逻辑资源的一个复用周期 T 内有 m 个时间片,即 $s_{11}^{(3)}=s_{ij}^{(3)}\cdots=s_{nm}^{(3)}=s_\alpha$,$1\leqslant i\leqslant m$,$1\leqslant j\leqslant n$.

(2) 所有逻辑资源的复用周期都是相同的,用固定周期 $T^{(3)}$ 表示,满足

$$T^{(3)} = m\cdot s_\alpha + c = m\cdot s_\alpha + m\cdot c_0 \tag{5}$$

(3) 任务组的完成时间为

$$t_{\mathrm{ms}}{}^{(3)} = (m\cdot s_\alpha + c)\cdot\left\lceil\frac{1}{n}\cdot\frac{\max(\mathrm{ur}_i.w)}{w_\alpha}\right\rceil \tag{6}$$

(4) 任务组平均响应时间为

$$t_{\mathrm{ar}}^{(3)} = \frac{1}{m}(m\cdot s_\alpha + c)\cdot\sum_{i=1}^{m}\left(\left\lceil\frac{1}{n}\cdot\frac{\mathrm{ur}_i.w}{w_\alpha}\right\rceil\right) \tag{7}$$

(5) 任务组的平均延迟率

$$r_{\mathrm{as}}{}^{(3)} = \frac{m\cdot s_\alpha + c}{s_\alpha} \tag{8}$$

(6) 资源利用率为

$$r_{\mathrm{u}}^{(3)} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\frac{w_\alpha}{\mathrm{pr}_j.v}\cdot\left\lceil\frac{1}{n}\cdot\frac{\mathrm{ur}_i.w}{w_\alpha}\right\rceil\right)}{n\cdot t_{\mathrm{ms}}^{(3)}} \tag{9}$$

公式(9)中的分子表示所有任务在所有逻辑资源上实际运行的时间和,即 n 个资源实际处理任务的总时间.分母表示 n 个资源在任务组执行期间总的运行时间,即 $n\cdot t_{\mathrm{ms}}^{(3)}$.

在策略3中,无论用户任务调度到何种逻辑资源上,在相等的时间片内会执行相等的任务量,这从用户角度看是公平的.但是同时也看到,小任务会很快完成,其分配占用的时间片空闲且得不到利用.另一方面,速度较快的逻辑资源在完成固定任务量后,可能还有余力,没有充分地利用时间片,造成了资源的浪费.为了充分利用资源,因此必须进一步完善时分复用策略.

策略4 不等时间片不等任务量的时分复用策略:该策略的主要思想是:如图2c所示,在每个复用周期内,云资源为每个任务分配不相等的时间片,使得每一个任务分配的时间片 s_{ij} 与该任务的工作量成

正比,大的任务得到长的时间片.因此,每个用户任务在运行一个时间片之后,被执行的任务量占自己总任务量的比例数相同,最终达到不同的任务在几乎相同的时间内完成,实现用户间的真正公平.每个云资源在一个时间片内执行的任务量 w_{ij} 与该资源的计算能力成正比,能够充分利用每个云资源的计算能力.

但是,对于同一个用户任务 ur_i,不同逻辑资源分配给该任务的时间片仍是相同的,即 $s_{ij}=s_{ik}$.最小任务时间片的确定与策略 3 中时间片 s_α 要考虑的因素是完全一样的,用 s_β 表示,即 $s_\beta=s_\alpha$.用户任务 ur_i 的时间片大小为 $s_{ij}=s_\beta\cdot\frac{\mathrm{ur}_i.w}{w_\beta}$,$j=1,2,\cdots,n$.其中,$w_\beta=\min\limits_{1\leqslant i\leqslant m}\mathrm{ur}_i.w$,即最小任务的工作量.用户请求 ur_i 在计算节点 pr_j 上执行,一个时间片内被执行 ur_i 的任务量为 $w_{ij}=s_{ij}\cdot\mathrm{pr}_j.v$.不同计算节点由于处理速度不同,一个时间片内执行的任务量也不同.

根据时分复用的原理,如图 2c 所示,策略 4 具有以下性能指标:

(1) 每个逻辑资源的一个复用周期内有 m 个时间片,即 $s_{ij}^{(4)}=s_\beta\cdot\frac{\mathrm{ur}_i.w}{w_\beta}$,　$1\leqslant i\leqslant m,1\leqslant j\leqslant n$.

(2) 所有逻辑资源的复用周期都相同,用固定周期 $T^{(4)}$ 表示,满足

$$T^{(4)}=\Big(\sum_{i=1}^{m}\mathrm{ur}_i.w\Big)\cdot\frac{s_\beta}{w_\beta}+c \tag{10}$$

(3) 任务组的完成时间 $t_{\mathrm{ms}}{}^{(4)}$ 为

$$t_{\mathrm{ms}}{}^{(4)}=T^{(4)}\cdot\frac{w_\beta}{s_\beta}\cdot\frac{1}{\sum\limits_{j=1}^{n}\mathrm{pr}_j.v} \tag{11}$$

(4) 任务组平均响应时间为

$$t_{\mathrm{ar}}{}^{(4)}=t_{\mathrm{ms}}{}^{(4)} \tag{12}$$

(5) 任务组平均延迟率为

$$r_{\mathrm{as}}{}^{(4)}=\frac{1}{m}\sum_{i=1}^{m}\frac{T^{(4)}}{s_{ij}^{(4)}} \tag{13}$$

(6) 资源利用率为

$$r_{\mathrm{u}}^{(4)}=1-\frac{c}{T^{(4)}} \tag{14}$$

3.4　时分复用策略的类型优先的综合使用算法 TMFM(type_matching_first_multiplexing)

在实际的云计算系统中,用户任务的类型,任务可划分粒度等是多种多样的,计算资源也常常只能适合某一类型的用户任务.前面给出的 4 种资源管理策略需要综合考虑计算环境与任务类型,根据前面的分析,给出一个资源类型与用户任务类型匹配优先,有效和综合利用各种时分复用策略的算法.

算法 1　类型匹配优先的综合时分复用算法

输入:用户任务请求 $\mathrm{URS}=\{\mathrm{ur}_1,\mathrm{ur}_2,\cdots,\mathrm{ur}_m\}$,云系统物理资源 $\mathrm{PRS}=\{\mathrm{pr}_1,\mathrm{pr}_2,\cdots,\mathrm{pr}_n\}$.

输出:设定复用周期和满足用户要求的时间片的逻辑资源 $\mathrm{LRS}=\{\mathrm{lr}_1,\mathrm{lr}_2,\cdots,\mathrm{lr}_n\}$.

第 1 步,按照用户的任务类型 $\mathrm{ur}\cdot\lambda$,将任务划分为任务子集 $C_1,C_2,\cdots,C_M$.M 是用户需求任务类型的个数.

第 2 步,按照资源的计算类型 $\mathrm{pr}\cdot\lambda$,将资源划分为资源子集 $P_1,P_2,\cdots,P_M$.

第 3 步,对 C_i 中的任务,按照可划分粒度 g,对任务进一步生成独占任务子集 C_{i0} 和可划分任务子集 $C_{i1},\cdots,C_{ik}$.

第 4 步,对 P_i 中的资源,按照计算类型和能力进行分簇 $P_{i0},P_{i1},\cdots,P_{ik}$,以便优先考虑执行相应的任务子集 $C_{i0},C_{i1},\cdots,C_{ik}$.

第 5 步,对每一类细分后的任务组 C_{i0},若任务数少于计算节点数,即 $m\leqslant n$,或者 $m\geqslant n$ 且长任务数远小于短任务的数量,则采用策略 2,否则采用策略 1.

第 6 步,对每一类细分后的任务组 C_{ij},$j\geqslant 1$,若任务组中的任务量相差不大,且 P_{ij} 的计算能力几乎相同的情况下,为了简化管理,则采用策略 3;其他情况下,则采用策略 4.

第 7 步,当所有组任务都执行完毕后,云管理系统进行下一批任务的调度执行.

4　资源时分复用策略理论分析评价和实验验证

4.1　资源时分复用策略的分析评价

沿用上述符号和假设,给出以下结论.

结论 1　策略 4 保证成批用户任务能够同时完成.

证明　因为按照策略 4,运行在 $\mathrm{LRS}=\{\mathrm{lr}_1,\mathrm{lr}_2,\cdots,\mathrm{lr}_n\}$ 上的任意任务 ur_i,其完成时刻 t_{fi} 满足

$$t_{fi}=T^{(4)}\cdot\frac{\mathrm{ur}_i.w}{\sum\limits_{j=1}^{n}w_{ij}}=T^{(4)}\cdot\frac{\mathrm{ur}_i.w}{\sum\limits_{j=1}^{n}(s_{ij}\cdot\mathrm{pr}_j.v)}=T^{(4)}\cdot\frac{w_\beta}{s_\beta}\cdot\frac{1}{\sum\limits_{j=1}^{n}\mathrm{pr}_j.v} \tag{15}$$

式(15)最右边与变量 i 无关,即所有用户任务的完成时间是相同的.证毕.

结论 2　策略 4 的性能优于策略 3,即:① $t_{\mathrm{ms}}{}^{(4)}$

$\leqslant t_{ms}^{(3)}$，② $t_{ar}^{(4)} \leqslant t_{ar}^{(3)}$，③ $r_{as}^{(4)} \leqslant r_{as}^{(3)}$，④ $r_u^{(4)} \geqslant r_u^{(3)}$.

证明由公式(5)，(6)和公式(10)，(11)可知

$$t_{ms}^{(3)} = \frac{m}{n} \cdot \frac{\max\limits_{1\leqslant i\leqslant m} ur_i.w}{\min\limits_{1\leqslant j\leqslant n} pr_i.v} + \frac{c}{n} \cdot \frac{\max\limits_{1\leqslant i\leqslant m} ur_i.w}{s_\alpha \cdot \min\limits_{1\leqslant j\leqslant n} pr_i.v},$$

$$t_{ms}^{(4)} = \frac{\sum\limits_{i=1}^{m} ur_i.w}{\sum\limits_{j=1}^{n} pr_i.v} + c \cdot \frac{\min\limits_{1\leqslant i\leqslant m} ur_i.w}{\sum\limits_{j=1}^{n} pr_j.v} \quad (16)$$

显然，在式(16)中，$t_{ms}^{(3)}$的第一部分与第二部分分别大于$t_{ms}^{(4)}$的第一部分与第二部分，所以有$t_{ms}^{(4)} \leqslant t_{ms}^{(3)}$.

同理可证$t_{ar}^{(4)} \leqslant t_{ar}^{(3)}$，$r_{as}^{(4)} \leqslant r_{as}^{(3)}$.

根据公式(9)和公式(14)，因为$u_r^{(4)}$仅有任务切换开销的影响，而$u_r^{(3)}$在策略3中短任务的时间片以及快计算节点的时间片都有部分浪费，易证$r_u^{(4)} \geqslant r_u^{(3)}$. 证毕.

结论3 ① 若$m \leqslant n$，即任务数小于等于机器数，则有$t_{ms}^{(1)} \geqslant t_{ms}^{(2)}$. ② 若$m > n$，且不考虑时分复用切换开销，则有$t_{ms}^{(2)} \geqslant t_{ms}^{(4)}$.

证明 根据定义5可知

$$t_{ms}^{(1)} = \max\left\{\frac{\min\limits_{i\in[1,m]} ur_i.w}{\max\limits_{j\in[1,n]} pr_j.v}, \cdots, \frac{\max\limits_{i\in[1,m]} ur_i.w}{\min\limits_{j\in[1,n]} pr_j.v}\right\} = \frac{\max\limits_{i\in[1,m]} ur_i.w}{\min\limits_{j\in[1,n]} pr_j.v}$$

$$t_{ms}^{(2)} = \max\left\{\frac{\max\limits_{i\in[1,m]} ur_i.w}{\max\limits_{j\in[1,n]} pr_j.v}, \cdots, \frac{\min\limits_{i\in[1,m]} ur_i.w}{\min\limits_{j\in[1,n]} pr_j.v}\right\}$$

显然，$t_{ms}^{(1)} \geqslant t_{ms}^{(2)}$.

根据公式(11)可知

$$t_{ms}^{(4)} = T^{(4)} \cdot \frac{w_\beta}{s_\beta} \cdot \frac{1}{\sum\limits_{j=1}^{n} pr_j.v} = \frac{\sum\limits_{i=1}^{m} ur_i.w}{\sum\limits_{j=1}^{n} pr_j.v}$$

$$t_{ms}^{(2)} = \max\left\{\frac{\max\limits_{i\in[1,m]} ur_i.w}{\max\limits_{j\in[1,n]} pr_j.v}, \cdots\right\} + \max\left\{\frac{\max\limits_{i\in[n+1,m]} ur_i.w}{\max\limits_{j\in[1,n]} pr_j.v}, \cdots\right\} + \cdots = \max\limits_{1\leqslant j\leqslant n}\left(\sum_{y=1}^{x_j} \frac{ur_{jy}.w}{pr_j.v}\right)$$

所以有$t_{ms}^{(2)} \geqslant t \geqslant t_{ms}^{(4)}$，证毕.

显然，由结论1、结论2、结论3可得如下结论.

结论4 极端情况$m=1$时，时分复用策略3、策略4较独占策略1、策略2的系统利用率和用户平均响应时间都极大缩短. 极端情况$n=1$时，时分复用策略3、策略4让用户使用资源的公平性充分体现，用户只需要等待最多一个复用周期T即可使用云资源，同时云系统利用率和用户平均响应时间都极大缩短.

进一步，还可得如下结论.

结论5 沿用上述符号，假设任务成批到达间隔为G，云系统调度周期为D，单批任务完成时间为t_{ms}，如果满足$t_{ms} \leqslant D$，则单批任务调度和资源管理策略是有效可行的；如果满足$m_{st} \leqslant D \leqslant G$，则整个云系统的任务调度和资源管理策略是有效可行的；如果满足截止时间要求，即$\lim\limits_{m\to\infty}(t_{ms} \leqslant \min\limits_{1\leqslant i\leqslant m} ur_i.\tau)$，则云系统PRS同时为用户集URS提供了无穷大的计算能力.

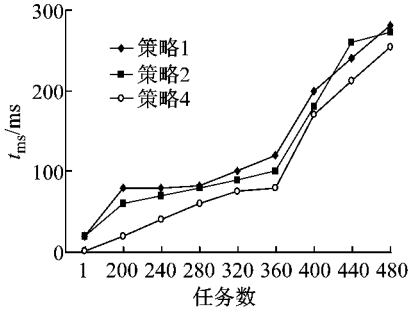
通过以上讨论可以看出：云资源时分复用虚拟化管理技术的本质是根据用户任务特征，以及资源环境状态，合理和科学地决定成批任务到达间隔G，云系统调度周期为D，单批任务完成时间t_{ms}，资源复用周期T_i，资源复用时间片s_{ij}等变量之间的关系，力求满足用户需求，满足结论5.

4.2 实验验证

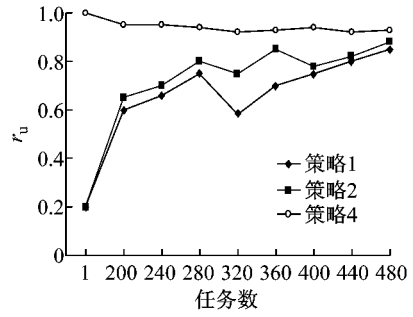
本文用CloudSim[16]来模拟云计算环境，CloudSim是由澳大利亚墨尔本大学的网格实验室和Gridbus项目推出的云计算仿真软件，是基于java的离散事件模拟工具包，支持云计算的资源管理和调度模拟. 本文选择CloudSim工具包进行模拟和评估本文调度算法是基于以下原因：① 能够建立和管理独立的任务；② 虚拟化能够在时间共享和空间共享之间灵活切换. ③ 提交给一个计算资源的作业数目没有限制；④ 支持静态和动态调度方式的模拟.

实验环境：一台Dell PC机，Intel 奔腾双核E5800，3.2 GHz，1GB DDR3. 安装的操作系统是Microsoft Server 2003，运行CloudSim的JDK为jdk1.6.0.

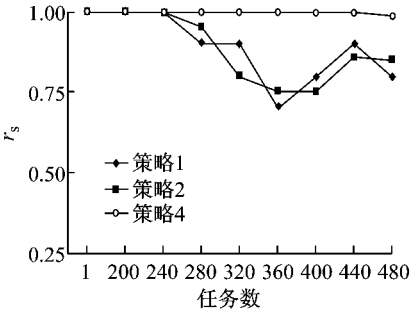
利用Cloudsim创建300台机器的云中心环境，图3给出了不同任务数目情况下的3种策略的性能比较，主要关注常用的任务组完成时间指标t_{ms}(图3a)，本文关注的资源利用率指标r_u(图3b)以及系统对任务组服务率指标r_s(图3c). 这里忽略策略3(因为策略3已经由结论2证明是策略4在同构机器环境下的一个简化处理版本). 实验任务数目从40到200，每隔40个任务采集1个样本.



a 完成时间指标



b 资源利用率指标



c 系统任务服务率指标

图 3　不同用户请求下策略的性能指标比较

Fig.3　Performance indicators under distinct user's requirements

从图 3a 可以看出，策略 4 的执行时间明显小于策略 1 和策略 2，特别是在任务数为 1 的时候，时分复用策略 4 可以利用整个系统的资源，而策略 1 和策略 2 只能使用一台资源. 在任务数小于机器数 300 的情况下，$t_{ms}^{(2)} < t_{ms}^{(1)}$，当任务数大于机器数 300 时，$t_{ms}^{(3)} > t_{ms}^{(4)}$，验证了结论 3 中的结论. 从图 3b 中可以看出，时分复用策略 4 对资源的利用率要明显高于其他两种策略，而图 3b，3c 则说明了随着任务数的增加，时分复用策略 4 能够服务更多的用户请求.

通过以上实验，看到本文提出的云资源时分复用策略在应对可划分批任务时，具有较高的资源利用率和较高的系统任务服务率，可以使得有限计算资源最大程度地应对用户任务执行需求. 而理论分析的结论 1 和结论 4 体现了系统对用户服务的公平性.

5　结语

本文针对云计算多租户环境，关注云计算中大量存在应用任务可切分的事实，考虑用户对于云资源公平性使用的需求，给出了一种时分复用的云资源虚拟化管理方法. 论文具体定义了云计算资源按需分配过程中涉及的物理资源、用户需求、逻辑资源等概念. 根据用户任务需求，以及资源的状态，通过设定复用周期 T 和时间片 s，将整个云计算资源进行了时分复用，形成虚拟资源，以便为每个用户提供时间片来运行任务. 针对不同的任务类型和可划分粒度等情况，提出了资源优先独占策略，相等时间片、相等任务量的时分复用策略，不等时间片、不等任务量的时分复用策略，以及类型匹配优先的综合时分复用算法. 最后，对不同策略性能指标进行了深入分析，形成了可以指导不同应用场景下的资源复用的相关结论：不可划分任务宜采用优先独占策略，可划分任务在工作量差别不大的情况下宜采用策略 3 简化管理，在复杂任务和资源情况下宜采用策略 4 等. 总之，本文提出的云资源时分复用方法，突出体现了云计算多租户环境下用户使用资源的公平性要求，使得有限计算资源最大程度地应对用户任务执行需求，同时大大提高云资源的利用率. 但是，本文假设任务是成批到达的，而不是随机到达；任务假设是可划分的独立任务，而不是一般的关联任务，假设条件比较理想化，后者情况非常复杂，笔者将在未来的工作中研究一般条件下云资源复用的方法，并且辅之以相应的实验验证.

参考文献：

[1] Foster I, Zhao Y, Raicu I, et al. Cloud computing and grid computing 360-degree compared[C]// Proceedings of IEEE Grid Computing Environments Workshop. Piscataway: IEEE Press, 2008:1-10.

[2] Armbrust M, Fox A, Griffith R, et al. A view of cloud computing[J]. Communications of the ACM, 2010, 53(4):50.

[3] Buyya R, Yeo C S, and Venugopal S. Market-oriented cloud computing: vision, hype, and reality for delivering IT service as computing utility[C]// Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications. Piscataway: IEEE Press, 2008:5-13.

[4] Zenoss Inc. Virtualization and cloud computing survey[EB/OL]. [2010-10-15]. http://www.zenoss.com/in/Virtualization_survey.html.

（下转第 803 页）