

开源云计算资源调度策略优化研究

邱亚¹, 颜金尧², 陈宇¹, 耿俊杰²

(1. 中央广播电视总台技术局, 北京 100020;

2. 中国传媒大学媒体融合与传播国家重点实验室, 北京 100024)

摘要:云计算技术是互联网服务模式的一种产物,具有大规模、可扩展、通用性强、可靠性高等特点。针对 OpenStack 云平台虚拟机调度管理机制进行研究,围绕虚拟机初始分配及运行阶段的资源调度策略存在的问题,对 OpenStack 云平台虚拟机调度管理策略进行了改进,以提高云计算平台资源利用率,改善物理主机负载均衡,进而保障用户云服务质量。该文优化了计算节点主机过滤机制,同时提出虚拟机运行阶段的动态迁移机制以及待迁移虚拟机的选择机制,保证了虚拟机初始分配的合理性以及运行阶段各计算节点物理主机的负载均衡。使用云计算仿真平台 CloudSim 对提出的优化策略进行了仿真。实验表明提出的虚拟机调度优化策略作用显著,实现了物理主机的负载均衡、提高了资源利用率,改善了平台服务质量。

关键词:云计算; OpenStack; 虚拟机调度; 负载均衡; 资源利用率

中图分类号: TP393.0

文献标识码: A

文章编号: 1673-629X(2023)06-0008-08

doi: 10.3969/j.issn.1673-629X.2023.06.002

Research on Optimization of Resource Scheduling in Open Source Cloud Computing

QIU Ya¹, YAN Jin-yao², CHEN Yu¹, GENG Jun-jie²

(1. Technology Bureau, China Media Group, Beijing 100020, China;

2. State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing 100024, China)

Abstract: Cloud computing technology is a product of the Internet service model, which has the characteristics of large scale, scalability, strong versatility and high reliability. We study the virtual machine scheduling management mechanism of the OpenStack cloud platform. In view of the problems existing in the initial allocation of virtual machines and the resource scheduling strategy in the running phase, the virtual machine scheduling management strategy of the OpenStack cloud platform is improved to improve the resource utilization rate of the cloud computing platform, improve the load balancing of physical hosts, and thus ensure the quality of cloud services for users. We optimize the computing node host filtering mechanism and propose a dynamic migration mechanism during the running phase of the virtual machine and a selection mechanism for the virtual machine to be migrated, which ensures the rationality of the initial allocation of virtual machines and the load balance of the physical hosts of each computing node during the running phase. We use the cloud computing simulation platform (CloudSim) to simulate the proposed optimization strategy. Experiments show that the virtual machine scheduling optimization strategy proposed has a significant effect, realizes the load balancing of physical hosts, improves resource utilization, and improves platform service quality.

Key words: cloud computing; OpenStack; virtual machine scheduling; load balancing; resource utilization

0 引言

近十几年来,云计算是一个热门话题。云计算服务正逐渐成为新型的信息基础设施,各个国家政府这

些年来纷纷鼓励厂商制定计划优先发展云服务,从而促进互联网技术不断创新^[1]。云计算是一种通过统一规划硬件系统资源,实现用户按需分配的技术,有利于

收稿日期: 2022-06-30

修回日期: 2022-10-30

基金项目: 国家重点研发计划(2019YFB1804303); 国家自然科学基金(61971382); 中国传媒大学高校基本科研业务费专项资金(CUC22CGJ02)

作者简介: 邱亚(1978-),男,高级工程师,硕士,研究方向为计算机网络存储、私有云技术;通信作者: 颜金尧(1973-),男,教授,博士,博导,研究方向为计算机网络、智能媒体技术。

资源的管理与监控^[2]。

OpenStack 作为目前流行的开源云计算管理平台项目,通过各个组件来完成云处理功能,覆盖了网络、服务器、操作系统、虚拟化等各个方面,提供了完整的 IaaS 解决方案^[3]。OpenStack 云计算平台成本低、可扩展性强、管理方便,成为企业认可的公有或私有云平台^[4]。

虚拟机是云平台提供服务的一种方式,资源的调度与管理就是通过对数据中心中各物理主机的资源进行监控,有效分配虚拟机的一个过程^[5]。在虚拟机运行当中,由于资源的实时变化,各个节点的规格差异,容易形成资源的不平衡分布。如今云计算虚拟资源的调度管理涉及到多个方向:提高资源利用率、降低能耗、提升经济效益或提高服务质量、改善任务调度性能等^[6]。

在任务调度方面,文献[7]优化了启发式异构算法(HEFT)有效解决了 workflow 调度的难题,降低了调度的能量消耗。文献[8]提出了一种基于遗传算法的云计算任务调度算法,划分任务级别的同时考虑调度时间以及成本,具有更好的调度性能。

在降低能耗方面,文献[9]提出了一种兼顾性能和能效的调度算法 GAHFET,采用有效比率识别并关闭一组无效的处理单元以减小能耗,可以用于在大型计算系统中调度云应用。文献[10]尝试通过预测工作负载来计算需求,修改可用服务器的集合来适应预测,为此开发了预测优化调度算法,大大降低了云计算平台的能耗。

在资源利用率方面,文献[11]考虑最新的启发式算法,研究了将高性能计算作业映射到资源,严格评估云计算工作的负载分配状态,提出资源感知负载均衡算法,提高了云计算平台的资源利用效率。文献[12]设计了一种基于基线神经网络技术的动态负载均衡策略,该策略将根据服务器的剩余负载容量对服务器进行动态分类,并将任务部署到服务器上,这可以最小化任务的总执行时间,并最大化服务器的资源利用率,同时平衡长期过程中云数据中心的负载。

然而,现有研究都没有真正考量云数据中心的实际运行状态给计算节点带来的压力问题,也没有涉及在资源变化的过程中动态评测虚拟机的迁移问题。因此,针对这些问题,该文借助 OpenStack 平台,对虚拟机调度模块进行优化,同时改进计算节点评价策略、选择策略,提出运行过程中虚拟机选择和动态迁移算法,并进行了实验验证。实验结果表明,提出的调度优化策略可以改善计算节点主机的资源均衡度,提升资源利用率和云服务质量。

1 OpenStack 虚拟机调度原理及问题

OpenStack 云平台始于 2010 年 Rackspace 和 NASA 的联合项目,是一系列软件的集合,旨在通过虚拟化网络、存储、计算资源来提供全面的云服务。OpenStack 提供的架构类似于亚马逊的 EC2 和 S3^[13]。OpenStack 完全开源,具有兼容性好、灵活性强等特点,模块化的设计和灵活的部署方式使得 OpenStack 成为众多企业的选择。

1.1 OpenStack 虚拟机调度原理

云计算平台的调度目标是虚拟机,所有的虚拟机从节点主机获取资源,因此云平台调度的最终要求是要保障计算节点主机之间的资源能够有效分配并且正常使用。

OpenStack 原生的调度引擎分为两个步骤:

(1) 计算节点过滤:首先通过查询数据库获取所有计算节点并对各个节点进行资源的统计。接着配置过滤器对所有的节点进行条件性过滤,删除不符合条件的计算节点并返回。在用户创建虚拟机时,由于用户的需求,虚拟机的规格不尽相同,当过滤器生效后,可能出现返回列表为空,这表示此虚拟机无法运行在现有节点中,创建流程随之结束。

(2) 权值计算:当过滤完不合格的计算节点后,由于各个节点剩余的资源不相同,就产生了节点选择的顺序。OpenStack 默认通过基于内存的权值计算来判断节点的先后顺序,主机剩余的内存越大,权值就越小^[14]。

除了需要考虑虚拟机创建初期的调度之外,云平台虚拟机管理还包含了运行时期的监管。在云平台正常运行过程当中,各虚拟机的工作状态会导致计算节点之间的工作负载产生差异。为了能够改善物理主机的负载状况从而保证虚拟机流畅运行,OpenStack 采用了迁移技术对计算节点进行有效调整。目前虚拟机迁移技术分为两种类型:静态迁移和热迁移。

1.2 OpenStack 虚拟机调度管理存在的问题

(1) 虚拟机分配问题。

虚拟机的分配决定虚拟机在创建和迁移时放置在哪一个计算节点,也就是寻求一个最佳的映射能够让虚拟机正常运行并且各计算节点之间的资源占用合理。从 Nova 组件的调度模块分析可知,OpenStack 在分配虚拟机时,默认根据节点主机的剩余内存进行选择^[15]。虚拟机的资源有内存、CPU、磁盘等,OpenStack 平台在分配虚拟机时采用贪心策略,首先选择剩余内存大的节点进行放置。基于内存的调度策略过于简单,这是因为创建新的虚拟机时,如果总是优先选择剩余内存大的节点,可能会导致低规格的虚拟机占用了

所有高性能的节点,从而高规格的虚拟机无法创建。

(2) 虚拟机运行问题。

云计算相关资源在实际的环境中是多变的,尽管在初始分配中计算节点达到理想的均衡状态,但是由于用户任务的差异会使不同节点中虚拟机的状态不一致,从而导致一段时间后,某些计算节点的负载迅速变化。一旦某一计算节点的负载升高到极限,将会严重影响节点中虚拟机的正常运行,甚至有可能导致虚拟机宕机。反之,如果某一节点的负载降低到很小并维持相当长的一段时间,就会使平台资源利用率过低,产生浪费。OpenStack 云平台虚拟机调度管理策略是一种静态的策略,对虚拟机运行过程中的动态变化的考虑不足^[16]。

2 系统模型

2.1 虚拟机调度管理优化建模

(1) 调度管理优化思路。

针对以上分析,OpenStack 平台在虚拟机分配和运行过程中的弊端主要有两个方面:

(a) 计算节点的选择的局限性。OpenStack 云平台在虚拟机初始分配和运行过程中迁移时,基于内存选择计算节点,并采用贪心算法优先选择节点剩余内存最大的主机。而虚拟机和节点主机涉及到的资源有多种,仅仅以内存作为选择依据不能保证多个虚拟机合理分配到对应的计算节点,容易导致计算节点之间的资源分配不均匀。

(b) 虚拟机调度策略的静态性。云平台计算资源时刻变化,需要周期性地对云平台资源进行有效监控。OpenStack 原生调度策略不能及时根据节点负载进行虚拟机的动态迁移,无法在实际的云环境中保证计算节点的负载均衡,容易对用户的云服务质量产生影响。

OpenStack 云平台具有高扩展性,为了改进虚拟机调度管理策略,需要从以下几个角度出发:

- 优化计算节点选择策略。在原有基于内存的选择策略上进行扩展,重写 nova-scheduler 调度模块并增加新的调度策略,将多种计算资源考虑在内,保证多种类型的虚拟机在创建或者迁移时能够分配到合适的节点中。

- 优化云平台资源监控模块。在虚拟机迁移之前,需要对云平台节点资源以及虚拟机自身资源进行有效监控,整合所有的资源使用状况,为虚拟机的迁移做准备。

- 优化虚拟机迁移策略。虚拟机迁移策略包括了三个方面,第一是何时进行迁移,第二是选取哪些虚拟机进行迁移,第三是迁移到哪个目标节点。

通过这三个角度实现对云平台虚拟机资源的动态管理,可以保证计算节点的负载均衡,有助于提高云服务质量。

(2) 计算节点工作负载模型。

工作负载代表了物理主机在接收任务后的工作状态,云计算环境中计算节点的工作负载需要时刻进行监控。虚拟机在创建或迁移时对节点的选择需要考虑到内存、CPU、磁盘容量之外,还需要对节点的工作负载进行考量,从而将一些负载不足的节点过滤掉,让虚拟机创建在更合适的物理主机上。

物理主机的工作负载主要受三个因素影响,分别是内存利用率、CPU 利用率、磁盘利用率^[17]。为了使工作负载的定义更加明确,该文对这些影响因素进行定义,从而得出计算节点的工作负载模型。

内存利用率用 M_{ut} 表示,计算节点主机的内存利用率为节点内虚拟机使用的内存和节点内存的比值的百分数。具体公式如式(1)所示。

$$M_{ut} = \frac{M_{used}}{M_{total}} \quad (1)$$

其中, M_{used} 表示计算节点中虚拟机已经使用的内存, M_{total} 表示计算节点主机的总内存。

CPU 利用率用 C_{ut} 表示, CPU 利用率为节点内虚拟机使用的 CPU 数和节点主机的总 CPU 数比值的百分数。需要注意的是,在 OpenStack 云平台中,节点主机的一个 CPU 可以虚拟成 6 个虚拟的 CPU。因此,具体公式如式(2)所示。

$$C_{ut} = \frac{C_{used}}{C_{total} \times 6} \quad (2)$$

其中, C_{used} 表示节点内虚拟机已经使用的 CPU 数, C_{total} 表示节点主机的 CPU 数。

同样,磁盘利用率用 D_{ut} 表示,磁盘利用率为节点内虚拟机使用的磁盘大小和节点主机总的磁盘大小的比值的百分数。公式如式(3)所示。

$$D_{ut} = \frac{D_{used}}{D_{total}} \quad (3)$$

其中, D_{used} 表示虚拟机占用的磁盘容量, D_{total} 表示节点主机的总磁盘容量。

节点主机的工作负载和以上三种因素有很大关系,该文用三者的平均值来描述计算节点的工作负载情况,这样能够更加明确地反映出节点主机的资源使用的实时性。用 Load 来表示计算节点的工作负载,具体公式如下:

$$Load = \frac{M_{ut} + C_{ut} + D_{ut}}{3} \quad (4)$$

(3) 虚拟机规格评价模型。

不同用户对云平台虚拟机的要求是不一样的。有

的虚拟机需要具备较多的 CPU 资源以满足计算性能,有的虚拟机需要具备较大的磁盘用来提高存储能力,还有一些虚拟机需要较大的内存资源。同样,云环境中计算节点通常也是异构的,各节点的性能也是不一样的^[18]。笔者希望通过对虚拟机进行评价并进行分类,区分的种类包括内存侧重型虚拟机、CPU 侧重型虚拟机以及磁盘侧重型虚拟机。在对虚拟机进行分配时,根据虚拟机的类型选择相应的权值计算器对节点主机进行排序,选择出适合放置该类型虚拟机的计算节点。

为了对虚拟机进行分类,首先需要定义一个基础的虚拟机类型。通过将用户需要的虚拟机参数和基础虚拟机的参数比较来衡量虚拟机的具体类型。定义的基础虚拟机参数如表 1 所示。

表 1 基础虚拟机参数

内存/G	CPU/核	磁盘容量/G
2	2	30

分别用 M_{base} 、 C_{base} 、 D_{base} 代表基础虚拟机的内存、CPU 和磁盘容量,用 M_{need} 、 C_{need} 、 D_{need} 分别代表用户需求的内存、CPU 和磁盘参数。同时,将用户的需求参数和基础虚拟机参数做比较,得出需求资源参数超出程度(设为 W)最大的设为侧重型资源。具体计算方式如式(5)~式(7)。

$$W_m = \frac{M_{need} - M_{base}}{M_{base}} \quad (5)$$

$$W_c = \frac{C_{need} - C_{base}}{C_{base}} \quad (6)$$

$$W_d = \frac{D_{need} - D_{base}}{D_{base}} \quad (7)$$

其中, W_m 、 W_c 、 W_d 分别表示用户虚拟机的内存、CPU、磁盘容量超出基础虚拟机的程度,值越大代表其超出基础参数的程度越大,找出三个值当中的最大的对应的资源就是虚拟机对应的侧重型资源。

(4) 计算节点负载均衡度模型。

基于 OpenStack 云平台虚拟机调度管理策略优化的最终目的之一是实现计算节点的负载均衡。计算节点的负载均衡是通过节点资源利用率进行评价,上文通过内存、CPU、磁盘这三种资源定义了节点主机的工作负载。如果用 $Load_i$ 表示第 i 台计算节点的工作负载,那么云计算中心计算节点的平均工作负载如公式(8)所示。

$$Load_{avg} = \frac{\sum_{i=1}^m Load_i}{m} \quad (8)$$

其中, m 为云计算系统中计算节点的个数,所有节点

的平均工作负载对于整个云平台系统的负载均衡度评价具有重要意义。负载均衡度的计算方式如式(9)所示。

$$H = \sqrt{\frac{\sum_{i=1}^m (|Load_i - Load_{avg}|)^2}{m}} \quad (9)$$

式中, H 表示云平台系统的负载均衡度,从理论上来看,当 H 越小时,计算节点之间的负载相差越小,说明主机之间的资源使用越均衡。

2.2 虚拟机调度管理优化实现

(1) 计算节点选择策略优化实现。

计算节点选择策略至关重要,它决定了云平台虚拟机的初始放置位置。OpenStack 云平台的节点选择策略分为节点过滤和权值计算,通过上文的分析以及相关模型的建立,该文提出改进策略。在节点过滤的过程中,在原有的基于内存的过滤机制上综合 CPU、磁盘容量过滤,同时加入计算节点工作负载过滤策略。也就是说,在虚拟机初始放置的过程中,不仅要避免分配到基础资源不足的节点,还要避免分配到工作负载不足的节点。

作为计算节点的物理主机在运行的时候,需要衡量运行状态是否正常。该文加入节点负载过滤策略之后,需要定义一个过滤的阈值,当节点主机的工作负载大于过滤阈值时,说明此节点主机负载达到极限,不适合再分配其它虚拟机。与此同时,还定义了运行阈值,当节点主机的工作负载超过运行阈值时,代表此节点主机的负载较高,需要时刻注意负载的变化。OpenStack 可以在配置文件中设置过滤阈值和运行阈值的大小。如设置过滤阈值为 90%,运行阈值为 80%。这样的设置表示当节点主机工作负载超过 90% 时,说明此计算节点不适合再分配虚拟机,当节点主机工作负载超过 80% 时,说明此节点的负载较高。

计算节点选择策略优化后,基本流程见图 1:

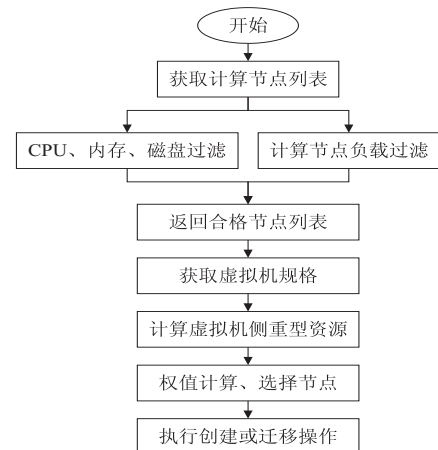


图 1 计算节点选择策略流程

(a) 获取所有计算节点列表。

(b) 调用相关类函数及重写的模块函数对计算节点进行基于内存、CPU、磁盘、工作负载的过滤, 去除不符合任一条件的节点。

(c) 返回合格节点列表。

(d) 获取创建或迁移的虚拟机规格。

(e) 对虚拟机进行侧重型资源的评价。

(f) 调用对应的权值计算器, 进行节点的权值计算并排序, 确定最终的计算节点。

(g) 执行虚拟机的创建或迁移操作。

权值计算是在节点过滤之后实现的, OpenStack 提供的权值计算机制就是将多个合格节点进行排序, 选出最佳节点完成虚拟机的最终创建^[20]。该文按照虚拟机的侧重型资源选择对应的权值计算器对计算节点进行过滤, 由此需要定义的权值计算器有三种, 分别是内存权值计算器、CPU 权值计算器和磁盘权值计算器。对于多个权值计算器的情况, 采用加权求和的方法来计算节点的最终权值。

通过对节点过滤策略和权值计算策略的重写, OpenStack 云平台不仅能够在虚拟机创建时全面衡量计算节点主机的资源状况, 还能在运行过程中为虚拟机迁移调用相关的节点选择机制。

(2) 虚拟机迁移策略优化实现。

虚拟机的分配策略是一种静态的调度策略, 云平台的负载状况在运行过程中会出现实时性变化, 该文对虚拟机迁移策略进行优化, 实现了一种动态调度策略, 根据计算节点主机的实时负载, 动态调整虚拟机所在的节点映射, 从而保证云平台能够负载均衡。

虚拟机迁移策略优化实现需要从三个方面入手^[19], 分别是迁移触发的临界点、待迁移虚拟机的选择和目标节点的选择。其中, 目标节点的选择在带迁移虚拟机确定之后可通过直接调用计算节点选择策略进行实现, 因此迁移策略的重点是确定迁移触发临界点以及选择需要迁移的虚拟机。

当虚拟机迁移的条件触发后, 需要选择告警队列中的节点作为被迁移节点, 对该节点上的虚拟机执行迁移操作。

虚拟机迁移的条件触发之后需要及时对相应的节点进行调整, 选出需要迁移的虚拟机并执行操作。待迁移虚拟机的选择需要考虑以下两个方面: (a) 虚拟机迁移时间: 尽管 OpenStack 能够在不中断用户服务前提下进行迁移, 但是在待迁移虚拟机选择时, 仍然期望选择迁移时间较短的虚拟机。(b) 服务质量: 节点主机的负载过高导致了服务质量下降, 因此在虚拟机迁移时, 需要考虑是否能够有效缓解主机负载状况。

结合计算节点工作负载的计算模型, 考虑将节点主机的内存使用率、CPU 使用率以及磁盘使用率进行比较, 找出使用率最大的作为节点主机最缺少的资源类型。当此资源类型确定后, 优先对相应资源占有率最大的虚拟机进行迁移。

综合以上两个因素, 定义了虚拟机迁移选择因子如公式(10)所示:

$$H_i = \frac{C_i}{M_i} \quad (10)$$

式中, H_i 代表了虚拟机对应节点主机最缺少的资源的占有率, M_i 代表虚拟机内存使用率。当虚拟机的 H_i 越大时, 应优先被视为待迁移虚拟机。

根据以上分析, 可以得出待迁移虚拟机选择策略的实现基本流程, 如图 2 所示。

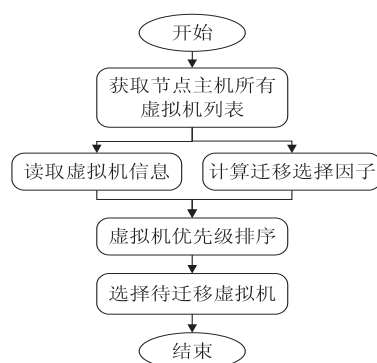


图 2 待迁移虚拟机选择策略

(a) 输入计算节点主机的 id 之后, 要获取该节点主机中所有的虚拟机列表。

(b) 对节点的资源使用率进行比较, 得出使用率最高的资源作为缓解负载的有效资源。

(c) 读取虚拟机信息, 确定内存使用率, 计算每台虚拟机的迁移选择因子。

(d) 根据迁移选择因子, 对虚拟机进行优先级的排序。

(e) 将优先级最高的虚拟机, 作为待迁移虚拟机。

(f) 结束流程, 将待迁移虚拟机传入其他迁移函数。

3 仿真实验

该文采用 CloudSim^[20] 来构建与 OpenStack 云平台同等类型的大规模云计算资源, 并对比优化后的虚拟机调度管理策略和 OpenStack 原生策略, 最后对实验结果进行分析。

3.1 实验设计

(1) 实验对象选取。

为了充分分析 OpenStack 虚拟机调度策略优化后的性能, 选取了另外两种资源调度策略进行对比实验,

分别是 OpenStack 默认的基于剩余内存的调度策略和一种随机的调度策略,具体说明如下:

- 基于剩余内存的调度策略:当创建虚拟机时,按照计算节点主机的剩余内存大小进行放置,优先选择剩余内存最大的主机。

- 基于粒子群算法的综合调度策略:该策略是现有成果中对 OpenStack 调度的一种综合改进策略,设计了适合云平台虚拟机分配的粒子群算法,并且采用了双阈值的迁移策略,从而一定程度上改进了云平台调度性能。

(2) 对比指标选取。

根据该文的优化目标,实验测试对比的指标主要有三个:节点负载、虚拟机迁移次数、资源利用率。

- 节点负载:建立了相应的节点负载模型,考虑到大规模计算节点群的总体均衡程度,通过定义节点的负载均衡度来进行总体评价,将优化的虚拟机资源调度策略与其它策略进行对比,得出优化策略的总体性能。

- 虚拟机迁移次数:虚拟机迁移次数体现了虚拟机初始分配后计算节点的负载变化带来的影响,因此虚拟机迁移的次数能够表现调度算法策略的优劣。

- 资源利用率:资源利用率体现了在保证云平台业务正常运行下对计算资源的使用率能够达到最高,从而不会造成平台资源的浪费。因此,最好的衡量手段是比较不同调度策略基础上能够创建的最大虚拟机数量。

(3) 计算节点设计。

真实的云计算环境要求计算节点之间存在规格差异,因此通过 CloudSim 模拟了异构的计算节点。为了测试提出的优化调度策略,需要构建大规模的计算节点群,在节点群中主要涉及三种类型:内存侧重型节点、CPU 侧重型节点和磁盘侧重型节点,总体规模为 90 台,每种类型的节点为 30 台。具体参数见表 2。

表 2 计算节点具体参数

计算节点	具体参数
类型	{ 内存侧重型, CPU 侧重型, 磁盘侧重型 }
内存	{ 64, 32, 8 }
CPU	{ 4, 8, 4 }
磁盘	{ 500, 500, 2 000 }

(4) 虚拟机设计。

同样,由于云用户对于业务的多样性需求,对平台中虚拟机规格的要求也是多样性的。该文将需要创建的虚拟机也分为三类:内存侧重型虚拟机、CPU 侧重型虚拟机、磁盘侧重型虚拟机,具体参数见表 3。

表 3 虚拟机参数

虚拟机	具体参数
类型	{ 内存侧重型, CPU 侧重型, 磁盘侧重型 }
内存	{ 8, 4, 2 }
VCPU	{ 2, 4, 2 }
磁盘	{ 50, 50, 200 }

3.2 实验和结果分析

(1) 节点负载对比。

为了比较不同资源调度策略给计算节点的节点负载带来的不同程度的影响,通过 CloudSim 创建了 90 台主机(参数如表 2)作为计算节点,三种类型的节点数量平均。同时,通过仿真软件随机定义了 150 个虚拟机用来分配。实验过程中,通过脚本分别实现了三种资源调度策略,包括该文优化的调度策略、OpenStack 默认基于内存的资源调度策略和基于粒子群算法的综合调度策略。

利用计算节点工作模型和负载均衡度模型对三种策略的分配结果进行定时计算,得到如图 3 所示的结果。

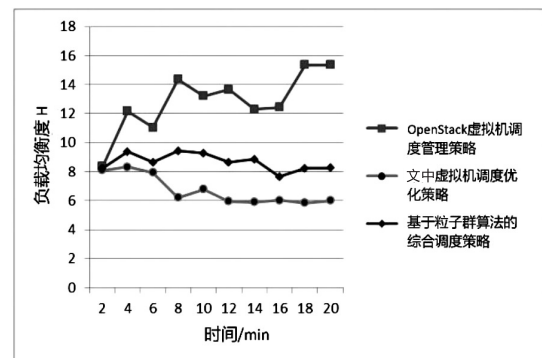


图 3 节点负载比较

从图中可以看出,不管是哪种资源调度分配策略,都会随时间造成整个云平台计算节点的总体负载均衡度的波动。首先可以明确看到 OpenStack 默认的基于内存的资源调度策略对云平台整体负载的影响,也就是计算节点总体负载均衡度的值在同一时间要大于其它两种策略。当负载均衡度越大时,表明节点的综合负载越不均衡。OpenStack 基于内存的调度策略会造成云平台负载不均衡是因为其本身运用的是一种静态的简单调度算法,这种算法在云平台实际运行中不能动态的改善节点负载。对比该文优化后的资源调度策略和基于粒子群算法的调度策略,两种策略下云平台的节点负载均衡度的总体变化趋势相似,但是该文的优化调度策略对云平台整体节点负载的改善更加明显。在优化目标的选择上,该文的调度策略充分考虑了影响计算节点的负载的多个因素,利用节点评价模型和虚拟机的动态迁移技术的优化,能够对云平台的

节点负载进行动态调整。

(2) 虚拟机迁移分析。

为了验证该文优化后的资源调度策略对云平台节点负载的改善效率,实验过程中将对比不同调度策略在云平台运行过程中导致的虚拟机迁移次数。需要说明的是,OpenStack 默认的虚拟机调度策略是静态的,不具备虚拟机动态迁移的功能,因此实验中只将该文优化的调度策略和综合调度策略做对比,具体情况如图 4 所示。

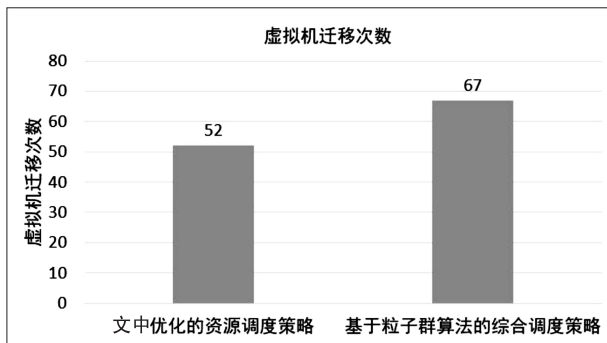


图 4 虚拟机迁移次数比较

从实验结果中可以看出,该文优化的资源调度策略在云平台运行过程中触发的虚拟机迁移次数要明显小于基于粒子群算法的调度策略,这是因为该文对虚拟机迁移策略做了一定的改进,分别在迁移触发的临界点、待迁移虚拟机的选择和目标节点的选择时综合考虑了迁移的时间和对服务质量的影响,所以避免了不必要的虚拟机迁移,有效地改善了云平台整体的负载。基于粒子群算法的调度策略虽然采用了双阈值的迁移机制,但是对待迁移虚拟机的选择是不够全面的,没有多方面的考虑迁移的成本。

(3) 资源利用率分析。

对于云平台的资源利用率的测试,比较了不同调度策略下虚拟机创建的最大数量。在 OpenStack 默认的调度策略中,在虚拟机初始创建时采用了简单的贪心算法。在基于粒子群算法的调度策略中,主要运用了粒子群算法的放置策略。在综合了节点工作负载模型和虚拟机评价模型,提出了计算节点选择策略。在实验中,在构建了 90 台计算节点的基础上,生成包含了三种类型的 500 台虚拟机,数量随机,接着用三种调度策略在云平台中不断创建虚拟机,直到不能创建为止。图 5 是三种调度策略的对比情况。

从实验结果可以发现,在同样的云环境下,该文的优化调度策略可以比 OpenStack 原生的调度策略创建更多数量的虚拟机。这是因为 OpenStack 原生的虚拟机调度机制比较简单盲目,只是优先选择剩余内存最大的节点,不能够实时考虑其它资源的限制。和基于粒子群算法的调度策略相比,该文优化的调度策略在

追求资源利用率的同时,还考虑了云平台的服务质量,因此不会盲目使得利用率过大而导致节点主机的工作负载变大。为了验证这一点,在达到创建虚拟机的最大数量时,分别测试了云平台计算节点的平均工作负载。

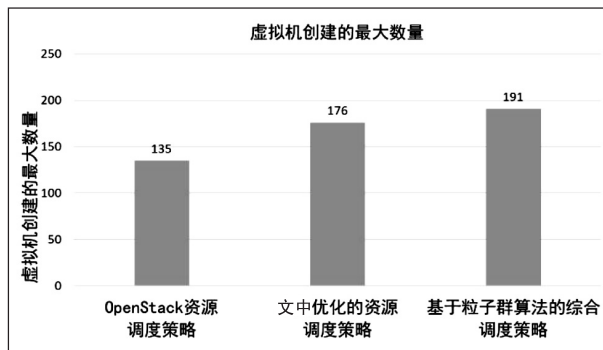


图 5 虚拟机最大迁移数量比较

图 6 中数据表明,该文的优化调度策略下计算节点主机的平均工作负载为 72.6%,而基于粒子群算法的调度策略下计算节点的平均工作负载为 89.8%。因此提出的优化调度策略能够在条资源利用率的前提下,最大程度地保证云服务的质量。

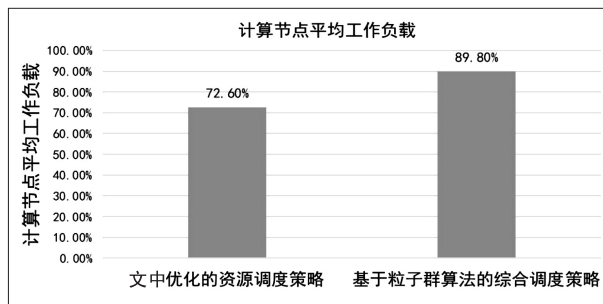


图 6 计算节点平均工作负载比较

4 结束语

以 OpenStack 云计算平台默认的虚拟机调度管理策略为基础,提出了具体的优化方案并实现了动态调度机制,提高了云平台资源利用率和节点的综合负载均衡。虽然这些优化方案经过实验得到了有效验证,但是仍然存在一些不足,需要在未来的一段时间继续研究。首先,继续研究物理主机的负载模型和虚拟机迁移策略,在优化虚拟机迁移策略时,通过计算主机一段时间的工作负载的平均值和负载阈值进行比较,在实际的环境中,云平台的运行通常具有一定规则,通过历史数据预测物理主机的负载,可以作为接下来的一个研究方向。其次,继续研究计算节点选择策略的优化目标,在计算节点选择策略上,选择了对内存、CPU、磁盘进行过滤,在实际的云环境中,计算节点主机的网络带宽等因素对主机也有一定影响,因此也将是接下来继续研究的方向。

参考文献:

- [1] 张琦. 云计算及关键技术的发展[J]. 计算机与网络, 2019, 45(24): 44.
- [2] 秦晋渊. 云计算技术发展趋势研究[J]. 电子测试, 2019(16): 123-124.
- [3] 魏迎. OpenStack 云计算平台的研究与实现[J]. 电子设计工程, 2019, 27(6): 152-155.
- [4] SEFRAOUI O, AISSAOUI M, ELEULDJ M. OpenStack: toward an open-source solution for cloud computing[J]. International Journal of Computer Applications, 2012, 55(3): 38-42.
- [5] BELOGLAZOV A, BUYYA R. OpenStack neat: a framework for dynamic and energy-efficient consolidation of virtual machines in OpenStack clouds[J]. Concurrency and Computation: Practice and Experience, 2015, 27(5): 1310-1333.
- [6] 李萌, 刘鑫. 基于改进 PSO 的云计算资源调度路径优化研究[J]. 计算机仿真, 2021, 38(9): 231-234.
- [7] DIVYAPRABHA M, PRIYADHARSHNI V, KALPANA V. Modified heft algorithm for workflow scheduling in cloud computing environment[C]//2018 second international conference on inventive communication and computational technologies (ICICCT). Coimbatore: IEEE, 2018: 812-815.
- [8] CUI Y, ZHANG X. Workflow tasks scheduling optimization based on genetic algorithm in clouds[C]//2018 IEEE 3rd international conference on cloud computing and big data analysis (ICCCBDA). Chengdu: IEEE, 2018: 6-10.
- [9] THANAVANICH T. Energy-aware and performance-aware of workflow application with hybrid scheduling algorithm on cloud computing[C]//2018 22nd international computer science and engineering conference (ICSEC). Chiang Mai: IEEE, 2018: 1-5.
- [10] DAMBREVILLE A, TOMASIK J, COHEN J, et al. Load prediction for energy-aware scheduling for cloud computing platforms[C]//2017 IEEE 37th international conference on distributed computing systems (ICDCS). Atlanta: IEEE, 2017: 2604-2607.
- [11] HUSSAIN A, ALEEM M, ISLAM M A, et al. A rigorous evaluation of state-of-the-art scheduling algorithms for cloud computing[J]. IEEE Access, 2018, 6: 75033-75047.
- [12] PAUL S, ADHIKARI M. Dynamic load balancing strategy based on resource classification technique in IaaS cloud[C]//2018 international conference on advances in computing, communications and informatics (ICACCI). Bangalore: IEEE, 2018: 2059-2065.
- [13] PRAKASH V, BAWA S, GARG L. Multi-dependency and time based resource scheduling algorithm for scientific applications in cloud computing[J]. Electronics, 2021, 10(11): 1320.
- [14] 毛军礼. OpenStack 之 Nova 服务[J]. 计算机与网络, 2018, 44(3): 60-63.
- [15] MORAIS F, FARIAS G, CARVALHO M, et al. Supporting mixed workloads in OpenStack-based clouds[C]//2018 IEEE 11th international conference on cloud computing (CLOUD). San Francisco: IEEE, 2018: 822-826.
- [16] WANG Z, ZHANG H, SHI X, et al. Efficient scheduling of weighted coflows in data centers[J]. IEEE Transactions on Parallel and Distributed Systems, 2019, 30(9): 2003-2017.
- [17] YAMATO Y, NISHIZAWA Y, NAGAO S, et al. Fast and reliable restoration method of virtual resources on OpenStack[J]. IEEE Transactions on Cloud Computing, 2015, 6(2): 572-583.
- [18] TOOSI A N, BUYYA R. Acinonyx: dynamic flow scheduling for virtual machine migration in SDN-enabled clouds[C]//2018 IEEE intl conf on parallel & distributed processing with applications, ubiquitous computing & communications, big data & cloud computing, social computing & networking, sustainable computing & communications (ISPA/IUCC/BD-Cloud/SocialCom/SustainCom). Guangzhou: IEEE, 2018: 886-894.
- [19] ZHAO H, ZHENG Q, ZHANG W, et al. Virtual machine placement based on the VM performance models in cloud[C]//2015 IEEE 34th international performance computing and communications conference (IPCCC). Nanjing: IEEE, 2015: 1-8.
- [20] 张梦. 基于 OpenStack 云平台的虚拟机资源调度方法研究[D]. 西安: 西安电子科技大学, 2017.