

未来网络试验设施的节点资源调度算法

汪晨欣 杨家海 庄奕 罗念龙

(清华大学网络科学与网络空间研究院 北京 100084)

摘要 随着互联网产业的扩张,对于网络核心技术的研究和创新刻不容缓,未来网络试验设施项目的建设为网络相关的科研人员提供高效便捷的试验环境,以支持网络技术的创新研究和实验。未来网络试验的基础设施资源是提供服务的基础,因此对试验资源的调度管理是项目中非常重要的任务。文中面向未来网络试验设施项目的资源调度和试验服务需求,设计了集中与分布相结合的架构,通过使中心资源调度管理系统与节点资源调度管理系统相互配合,来协调调度主干网带宽资源和位于各站点数据中心的资源。并且针对试验设施的特点,设计了综合考虑虚拟机间的通信代价、站点内物理机的平均资源利用率和资源均衡的多目标优化节点资源调度算法。仿真实验结果表明,该算法能有效实现上述多个目标的优化。

关键词 多节点分级调度架构,多目标优化,虚拟机资源调度

中图分类号 TP302 **文献标识码** A **DOI** 10.11896/jsjcx.190400106

Node Resource Scheduling for Future Network Experimentation Facility

WANG Chen-xin YANG Jia-hai ZHUANG Yi LUO Nian-long

(Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing 100084, China)

Abstract With the expansion of the Internet industry, research and innovation of network core technologies are urgent. The Future Network Experimentation Facility (FNEF) was designed to provide efficient and convenient resources for network-related researchers to support innovative research and experiments on network technologies. The resources scheduling management system is a very important task in FNEF. Based on the resource scheduling and test service requirement of FNEF, this paper designed a combination of centralized and distributed architecture. The central resource scheduling management system cooperates with the node resource scheduling management system to schedule the backbone network resources and computing resources in each disperse site. This paper proposed a multi-objective optimized resource scheduling algorithm by considering the communication cost between the virtual machines, the physical machine resource utilization and the resource balance. Simulation experiments show that the proposed algorithm can effectively optimize the above multiple objectives.

Keywords Multi-node hierarchical scheduling architecture, Multi-objective optimization, Virtual machine resource scheduling

1 引言

随着互联网产业的快速扩张,对网络性能、安全性等的要求不断提高,面向网络核心技术突破、网络性能提高、新型网络体系结构等的研究一直是热点和重点;然而科研环境下有限的试验资源和网络试验规模限制了研究和试验的进行。为了解决当前互联网发展所面临的问题,面向未来网络前沿科学技术研究,国家提出了中长期规划的国家重大科技基础设施建设项目——一个开放、易使用、可持续发展的大规模通用未来网络试验设施(Future Network Experimentation Facility),为互联网可持续发展提供简单、高效、低成本的基础理论

验证和关键技术研究的试验验证平台。

未来网络试验设施的资源包括主干网核心节点互联所需的带宽资源,以及分布在 40 个站点数据中心内的计算及存储资源。科研人员通过申请分布于不同站点数据中心的虚拟机和主干网网络资源,组建各自的大规模分布式 Overlay 试验网络环境,进行网络相关试验。合理地调度管理试验资源,在满足用户资源需求的条件下优化资源调度的性能是本项目的重要任务。因此,本文首先基于未来网络试验设施的需求,设计了集中与分布相结合的资源调度与试验服务系统架构以及分级调度流程,实现了全网资源的统一调度管理。其次,针对节点资源调度管理系统,为了减少站点数据中心的带宽资源

到稿日期:2019-04-18 返修日期:2019-07-31 本文受未来网络试验设施项目资助。

汪晨欣(1994—),女,硕士,主要研究方向为云计算、资源调度、网络管理,E-mail:wangcx16@mails.tsinghua.edu.cn;杨家海(1966—),男,教授,博士生导师,CCF 会员,主要研究方向为互联网络管理、网络测量与安全、云计算与虚拟化等,E-mail:yang@cernet.edu.cn(通信作者);庄奕(1995—),男,硕士,主要研究方向为云计算、资源调度;罗念龙(1971—),男,博士,副研究员,主要研究方向为计算机应用、数据分析。

浪费,提高资源利用率,降低运营成本,本文综合考虑资源需求、虚拟机间通信代价、资源利用率、资源均衡等多个因素,设计了多目标优化的节点资源调度算法,重点解决了站点数据中心内的资源调度问题。最后通过仿真实验验证了算法的有效性。

2 相关工作

2.1 GENI项目

GENI(Global Environment for Network Innovations)是由美国国家科学基金会赞助支持建设的网络试验设施平台^[1],为美国的下一代互联网科研人员提供了在现有网络及设备的基础上进行相关研究的开发和测试环境。GENI的管理核心层 GMC(GENI Management Core)^[2]定义了一个稳定、可预测和安全的框架,将物理资源抽象为组件、切片和集合的概念,来支持用户进行可扩展的试验和资源请求。虽然未来网络试验设施项目的目标与 GENI 项目有相似性,但两者在设计上有各自的独特性和创新性,同时 GENI 的技术细节并未对外公开。因此,需要针对未来网络试验设施自身的特点和需求,设计合适的资源调度与试验服务系统。

2.2 数据中心虚拟机资源调度

未来网络试验设施的站点数据中心,以提供针对网络试验定制化的虚拟机的形式向用户提供服务,用户可以根据试验需求申请不同资源配置的虚拟机。因此,对于节点资源调度管理系统而言,试验资源是以虚拟机资源的形式进行调度分配的,目前的研究工作主要对云计算环境下虚拟机调度的不同目标和侧重点进行优化^[3-6],包括用户服务质量 SLA、数据中心能耗、负载均衡等不同关注点^[7]。作为免费提供试验资源的基础设施平台,其在保证资源需求的基础上,将降低站点的运营成本作为重点考虑的目标;同时由于试验平台的特性,试验用户在站点内申请的虚拟机间需要持续通信和稳定带宽的场景是非常常见的,由此产生的通信消耗为站点内紧缺的带宽资源带来了很大挑战。因此,降低虚拟机间的通信代价,同时通过提高站点内物理机的平均资源利用率和资源均衡来减少资源浪费、降低运营成本是节点资源调度算法主要关注的优化目标。

针对虚拟机调度的通信代价优化,Alicherry等^[8]提出了分布式云中基于“network-aware”的虚拟机调度算法,在保证用户对虚拟机的带宽需求的同时,以最小化 VM 间的最大通信延迟为优化目标,但该算法没有考虑资源异构的情况。Shetty等^[9]在前者的基础上提出了一种考虑数据中心网络分层(树型)拓扑特点,支持 VM 异构的虚拟机调度算法。该算法将虚拟机分配优化抽象为最小高度子树问题,并利用启发式算法优化虚拟机间的通信代价。Canali等^[10]提出了综合考虑通信代价和能耗的虚拟机调度模型,并通过最小化虚拟机占用物理机的数量来降低能耗。Cohen等^[11]提出了一种在共享存储的云计算场景下优化虚拟机到数据中心单点(称为 root)的流量的近似算法。

针对资源利用率和负载均衡的优化,Tian等^[12]提出了预分区的方式,通过主动在各个物理机上提前做虚拟机的迁移

准备,来提高负载均衡算法的效率与稳定性,但由于资源副本会同时有多台物理机上准备,因此会影响物理机的有效利用率。Nagpure等^[13]利用偏度来衡量服务器资源利用率的不均匀,并通过负载预测来优化资源利用率。Zhang等^[14]基于改进 MOGA 算法的 MOGA-THSA 算法,对 SLA 违规率和负载均衡两个目标进行优化。Fan等^[15]提出了一种以降低服务器负载偏差度为目标的模拟退火负载均衡调度算法。

综上所述,在云计算环境下,资源调度的研究工作主要采用最小化虚拟机间最大通信延迟的策略来优化通信代价,但没有评估调度结果对数据中心整体通信代价带来的影响,存在一定的局限性。另外,未来网络试验设施是以科研为目的的免费资源提供平台,提高其资源利用率和资源均衡也是调度算法的重要目标,然而目前缺少综合考虑上述多个优化目标的研究工作。因此,本文提出了降低虚拟机间通信代价,同时提高站点平均资源利用率和资源均衡的多目标优化节点资源调度算法。

3 资源调度与试验服务系统

3.1 系统整体架构

本文基于未来网络试验设施的特点和需求,设计了资源调度与试验服务系统(以下简称为系统)的整体架构,如图 1 所示。

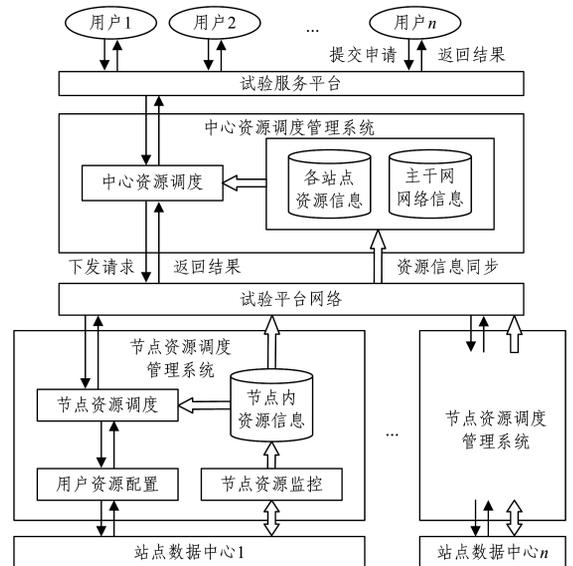


图1 系统整体架构

Fig. 1 System architecture

系统采用集中与分布相结合的设计框架,通过中心资源调度管理系统与分布式的节点资源调度管理系统协调配合,实现全网试验资源的调度管理,向各种试验用户提供试验资源和服务。系统主要由 3 个部分组成。

1) 试验服务平台:通过统一的用户服务界面,向试验用户提供登记注册和资源管理功能;接收试验用户的资源申请,将其提交给中心资源调度管理系统,并最终将资源调度结果返回给用户。

2) 中心资源调度管理系统:根据试验用户提交的跨节点资源需求,统一调度和管理试验设施的主干网网络资源,与

点资源调度管理系统配合、协调以调度位于各站点数据中心的物理资源。

3) 节点资源调度管理系统: 部署于分布式的各个站点数据中心内, 调度和管理本站点内的试验资源, 为本地用户提供站内试验服务; 同时配合中心资源调度管理系统, 为跨节点用户按需提供试验资源。

3.2 面向试验用户的资源分级调度

本文根据中心和节点协调配合的资源调度管理系统架构, 设计了面向试验用户的资源分级调度流程, 分阶段协作完成资源请求。资源分级调度流程如图 2 所示。

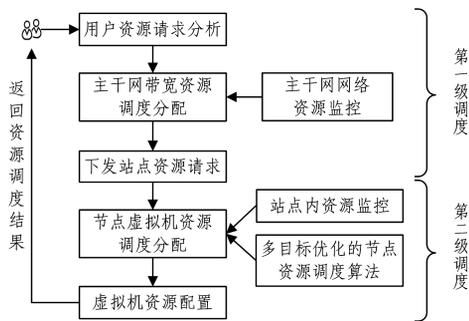


图 2 资源分级调度流程

Fig. 2 Process of hierarchical resource scheduling

(1) 第一级调度: 中心资源调度管理系统根据全局网络资源情况, 结合用户的带宽资源请求, 采用合适的调度算法分配主干网网络资源。由于项目的合作分工, 一级调度算法的具体设计并非本文的研究重点, 在此不再赘述。

(2) 第二级调度: 节点资源调度管理系统根据中心下发的用户虚拟机请求, 进行站点内虚拟机资源的调度分配, 采用本文设计的多目标优化节点资源调度算法, 选择站点内合适的物理机进行资源分配, 具体将在第 4 节进行描述。

集中与分布相结合的架构和资源分级调度流程的设计, 保证了调度系统的可扩展性和灵活性, 同时实现了中心资源调度管理系统与节点资源调度管理系统的松耦合, 通过分工配合实现资源的统一调度管理, 有利于系统模块的独立开发和未来扩展。

4 站点内虚拟机资源调度

根据前文介绍的系统架构, 节点资源调度管理系统的首要任务是进行站点内虚拟机资源调度, 即选择合适的物理机进行虚拟机资源分配。

4.1 站点内资源调度建模

本文定义站点数据中心的物理机资源集合为 $PM = \{pm_1, \dots, pm_i, \dots, pm_n\}$, n 表示物理机数量; 试验用户的虚拟机资源请求集合为 $VM = \{vm_1, \dots, vm_j, \dots, vm_m\}$, m 表示虚拟机请求数量。物理机和虚拟机包括 CPU、内存和带宽 3 种资源属性。基于未来网络试验设施的规模和所提供服务的特性, 资源调度考虑如下优化目标。

(1) 虚拟机间的通信代价。网络资源是站点数据中心的稀缺资源, 本文定义站点虚拟机间的通信消耗为:

$$C = \sum_{i \neq j} f_{ij} h_{ij}, i, j \in \{1, 2, \dots, m\} \quad (1)$$

其中, f_{ij} 表示虚拟机 vm_i 与 vm_j 间的通信流量, h_{ij} 表示部署 vm_i 和 vm_j 的物理机间的通信代价。本文将两个物理机间通信所需要经过的最少链路数作为通信代价。显然, 通信代价越大, 虚拟机之间通信消耗的网络资源就越多, 进而影响了站点数据中心的网络资源利用率, 降低了用户体验与平台稳定性, 增加了运营成本。因此, 可以通过降低通信代价来优化虚拟机间的通信消耗。

(2) 资源利用率。利用综合利用率乘积的方法^[16]定义站点内物理机的平均资源利用率, 具体公式为:

$$U = \frac{1}{n} \sum_{i=1}^n \frac{1}{(1 - u_i^{cpu}) \times (1 - u_i^{mem}) \times (1 - u_i^{bw})} \quad (2)$$

其中, n 表示物理机的个数; u_i^{cpu} , u_i^{mem} , u_i^{bw} 分别表示物理机 pm_i 的 CPU 利用率、内存利用率和带宽利用率。U 值越大, 站点内物理机的平均资源利用率就越高。

(3) 资源均衡。物理机不同资源的利用率不均衡会导致其中的紧缺资源影响到其他资源和整体的利用率, 产生短板效应, 进而导致物理机资源浪费, 增加站点运营成本。本文利用标准差的方式计算物理机 pm_i 的资源不均衡度, 具体公式为:

$$v_i = \sqrt{\frac{(u_i^{cpu} - u_i^{avg})^2 + (u_i^{mem} - u_i^{avg})^2 + (u_i^{bw} - u_i^{avg})^2}{3}} \quad (3)$$

$$u_i^{avg} = \frac{u_i^{cpu} + u_i^{mem} + u_i^{bw}}{3} \quad (4)$$

其中, u_i^{avg} 表示物理机 pm_i 的 CPU 利用率 u_i^{cpu} 、内存利用率 u_i^{mem} 和带宽利用率 u_i^{bw} 的平均值。v_i 值越小, 物理机不同资源的利用率就越均衡。

综上, 节点资源调度可以抽象为将给定的虚拟机请求集合映射到物理机集合的过程, 需要考虑的优化目标可表示为: $\min C$ and $\max U$ and $\min V$ 。另外, 为了保证虚拟机的性能, 定义物理机各项资源的利用率阈值为 T_{ucpu} , T_{umem} 和 T_{ubw} , 因此节点资源调度算法的约束条件是物理机的各项资源利用率不能超过阈值。

4.2 数据中心内物理机簇的抽象

为了对虚拟机间的通信代价进行优化, 本文算法抽象并定义了物理机簇的概念。

定义 1(物理机簇) 站点数据中心内一台交换机下所有叶子结点连接的物理机组成的集合(简称为簇)。以该交换机为该物理机簇的唯一标识, 表示为 $Cluster_id$ 。

簇内定义以下两个属性。

定义 2(簇代价) 一个簇内任意两台物理机间最大的通信代价, 表示为 C_cost 。

定义 3(簇利用率) 一个簇内所有物理机的平均资源利用率, 利用式(2)计算, 表示为 C_utili 。

根据定义可以将数据中心内的物理机资源以簇的形式进行抽象和管理, 每个簇表示为包含一组物理机和两个属性值的多元组, 其形式如下:

$$Cluster_id: (\{pm_i, pm_j, \dots\}, C_cost, C_utili) \quad (5)$$

假设某个站点数据中心内的网络拓扑如图 3 所示, 采用基于节点遍历的算法来初始化物理机簇, 结果如表 1 所列。

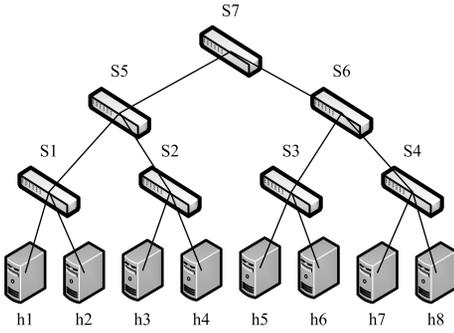


图3 数据中心树型网络拓扑

Fig. 3 Tree network topology in data center

表1 数据中心初始化后的物理机簇

Table 1 Clusters in data center

簇标记(ID)	物理机簇表示
S1	({h1, h2}, 2, u1)
S3	({h5, h6}, 2, u3)
S5	({h1, h2, h3, h4}, 4, u5)
S7	({h1, h2, h3, h4, h5, h6, h7, h8}, 6, u7)
...	...

在对数据中心进行物理机簇的抽象与初始化后,调度虚拟机时会基于物理机簇进行调度域选择。选择簇代价最小的物理机簇作为调度域能够有效优化通信代价。

4.3 节点资源调度算法

基于对站点内资源调度的分析与簇的定义,本文在站点数据中心典型的树型网络拓扑的基础上,设计了针对虚拟机间通信代价、资源利用率和资源均衡的多目标优化节点资源调度算法。该算法的调度过程主要分为两个步骤。

步骤1 基于物理机簇的虚拟机调度域的选择。

根据簇的定义,对数据中心的簇进行初始化并根据通信代价与利用率进行优先级排序,优先选择通信代价小的且簇利用率低的簇作为后续算法的虚拟机调度域。

此步骤通过初始化数据中心、划分物理机簇的方式来降低虚拟机间的通信代价。当有多个满足要求的簇时,以簇的利用率为优先级标准,利用率低的簇优先级高,以此改善数据中心整体的资源利用率和物理机簇间的负载均衡情况。

步骤2 簇内进行启发式虚拟机调度。

选定最合适的簇后,从簇内选取合适的物理机分配虚拟机。簇内物理机的平均资源不均衡度表示为簇内 k 台物理机不均衡度的平均值:

$$V = \frac{1}{k} \sum_{i=1}^k v_i \quad (6)$$

簇内进行虚拟机调度的过程是将虚拟机请求集合映射到簇内物理机集合的过程,求解优化目标值的最小映射方案。虽经过步骤1选取的簇调度域比以整个数据中心为调度域的规模小,但枚举比较每种可能的调度方案仍需很大的计算量。这种情况下,使用启发式算法能在可接受的时间和空间内得到近似最优解,因此本文采用改进的遗传算法(GA)来求解使不均衡度最低的虚拟机分配方案。

传统的遗传算法使用二进制编码方式对染色体进行编码,模拟生物进化的过程进行最优解的进化和搜索,本优化问题需要得到的最优解序列如下:

$$(vm_{a1}, vm_{a2}, \dots, vm_{am}), vm_{ai} \in \{pm_1, pm_2, \dots, pm_n\}$$

其中, vm_{ai} 表示虚拟机 vm_i 被分配到的物理机。本文对表示可行解的染色体进行编码,一条染色体包含 m 个基因位,每个基因位代表一台待分配虚拟机,每个基因位的编码值为调度结果选取的物理机的编号值。使用优化目标不均衡度 V 作为遗传算法的适应度函数。同时,在目标优化时虚拟机调度需要关注请求的响应时间,过大的种群空间和进化代数会影响算法的运行性能。为了在更短的时间内获得最优解,本文算法对传统遗传算法进行优化。首先,在生成初始种群时加入初代择优策略,从多次随机生成的个体中选择具有较好适应度值的个体作为初始种群,加快遗传算法的收敛速度,提高运行性能。其次,在种群进化过程中,为了保证优良个体的基因属性不被交叉和变异操作破坏,加入最优个体保留策略,将每一代中适应度最好的个体直接遗传到下一代,以提高解的优良性。

综上,节点资源调度算法的伪代码如算法1所示。

算法1 节点资源调度算法

输入:虚拟机请求序列 $vmList$; 站点数据中心物理机序列 $pmList$; 站点数据中心网络拓扑 $netTopo$; 物理机簇序列 $clusterList$; 簇内遗传算法参数: 种群大小 $popSize$, 进化代数 $genNum$, 交叉概率 $pCross$, 变异概率 $pMutation$

输出:虚拟机调度结果序列 $resultList$

1. 初始化 $clusterList$;
2. for switch in $netTopo$ do
3. 遍历交换机下叶子结点的物理机
4. 初始化此交换机表示的簇($\{pm\}, C_cost, C_util$)
5. insert $clusterList$
6. end for
7. for vm in $vmList$ do
8. 基于通信代价和利用率选取物理机簇;
9. 对 $clusterList$ 进行优先级排序;
10. 选取最优 $cluster$ 作为选取的调度域;
11. 执行簇内启发式虚拟机调度;
12. 初代择优策略生成初代 $popSize$ 个个体;
13. 计算每个个体的适应度 $fitness$;
14. for i in $genNum$ do
15. 种群选择操作
16. 以 $pCross$ 概率进行对点交叉操作
17. 以 $pMutation$ 概率进行多点变异操作
18. 保留 $fitness$ 最好的个体
19. end for
20. 记录调度结果并更新簇内属性值
21. end for
22. return $resultList$

4.4 仿真实验和结果分析

本文在 CloudSim 仿真平台上进行了多目标优化的节点虚拟机调度算法的模拟实验。基于 CloudSim 平台构建了如图3所示的树型网络拓扑的站点数据中心,边缘交换机总共连接了200台物理机。首先,为了仿真资源异构的情况,模拟了4种不同资源配置的物理机,每种配置50台,如表2所列。其次,为了模拟真实的虚拟机请求,本实验随机生成不同资源配置的虚拟机请求。按照表3所列的配置随机生成500个虚拟机请求进行增量式调度实验。

表 2 物理机资源配置

Table 2 Configuration of physical machine

分组	CPU/MIPS	内存/GB	带宽/Gbps	数量
1	1000	4	1	50
2	1500	8	1	50
3	2000	10	2	50
4	2500	16	2	50

表 3 虚拟机请求的配置

Table 3 Configuration of virtual machine requests

资源种类	资源配置
CPU/MIPS	150,200,250,300,350,400
内存/MB	500,600,800,1000,1200,1400
带宽/Mbps	50,100,150,200,250,300

为了分析本文中多目标优化节点资源调度算法的性能,在相同的实验环境中进行了两组实验。由于本文算法针对未来网络试验设施的特性对资源调度问题进行了分析建模,从而进行调度性能的优化,因此,首先将本文算法与实际云计算环境中较为通用和常见的调度算法进行对比,将其作为第一组实验,以测试本文算法相较于这些算法对优化目标的实际优化情况。另外,为了验证本文算法基于物理机簇的调度策略的优良性,将本文算法与一些研究工作中提出的对多个优化目标计算权重值,并利用启发式算法进行优化的方法进行第二组对比实验。具体实验结果分析如下。

(1)第一组实验。将本文算法与云计算环境下较常见的 First-Fit 和 Best-Fit 算法,以及 OpenStack 中默认使用的 RAMWeigher 算法进行虚拟机调度的对比实验。其中,First-Fit 算法在调度时选择第一个满足虚拟机资源需求的物理机进行分配;Best-Fit 算法则选择资源剩余最多的物理机进行虚拟机分配;RAMWeigher 算法根据物理机可用的内存资源计算物理机的权重,在调度时选择权重最大的物理机进行分配。为了对比这 4 种调度算法的性能,根据本文考虑的优化目标,利用式(1)、式(2)、式(6)分别在完成不同虚拟机请求数量时,计算调度后的虚拟机间的通信代价值、数据中心平均资源利用率值和平均资源不均衡度值,对比结果如图 4—图 6 所示。

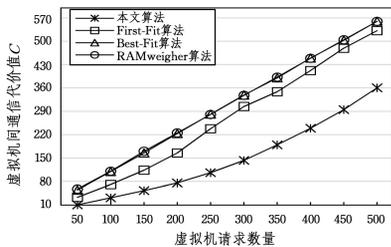


图 4 不同调度算法下虚拟机间的通信代价值 C 的对比

Fig. 4 Comparison of communication cost between virtual machines

in different scheduling algorithms

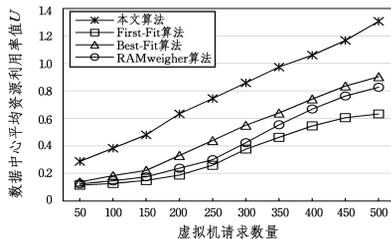


图 5 不同调度算法下资源利用率 U 的对比

Fig. 5 Comparison of resource utilization in different scheduling algorithms

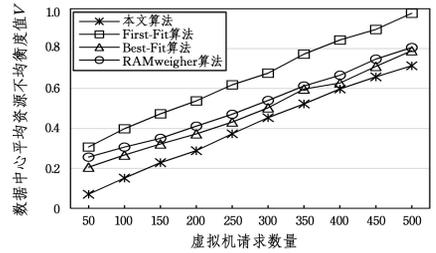
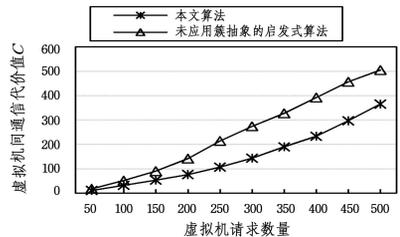


图 6 不同调度算法下资源不均衡度值 V 的对比

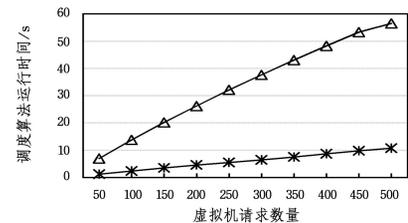
Fig. 6 Comparison of resource balance degree in different scheduling algorithms

从图 4 可以看出,本文算法对虚拟机间的通信代价做出了有效的优化。实验数据显示,本文算法中基于物理机簇的调度域选择策略相比其他算法,将通信代价平均降低了 35%。从图 5 可以看到,本文算法在基于簇代价选取调度域时综合考虑了簇利用率,优先选择利用率低的簇,该策略能够提高站点数据中心整体的平均资源利用率,从而在一定程度上改善负载不平衡的情况。从图 6 可以看到,本文算法在簇内进行调度时,采用了以资源均衡度为优化目标的遗传算法,相比其他算法也有明显的优化,尤其在本实验资源异构和具有一定随机性的情况下。而 First-Fit 和 Best-Fit 算法基于贪心的思想,缺乏对全局优化的考虑,因此容易得到局部最优解;RAMWeigher 算法虽然考虑了物理机各项资源的权重,但对权重的评价策略较为简单,对单一目标如内存资源利用率的优化效果较好。

(2)第二组实验。为了验证物理机簇的抽象和选取对通信代价优化的重要性,对未应用物理机簇抽象的启发式算法与本文算法的通信代价的优化情况进行对比。其中,未应用物理机簇抽象的启发式算法基于一些研究工作对多目标资源调度问题求解的思路,将本文考虑的多个优化目标值利用加权的方法映射为一个目标函数进行最优化问题求解。使用相同的参数和仿真环境进行虚拟机调度实验,对比调度后虚拟机间的通信代价值 C 和算法的运行时间,结果如图 7 所示。



(a)不同算法调度结果中虚拟机间的通信代价值对比



(b)调度算法运行时间对比

图 7 算法结果对比

Fig. 7 Comparison of scheduling results

从图 7(a)可以看出,本文提出的算法中,基于物理机簇的抽象和基于簇代价的调度域选择,能够有效地降低虚拟机

间的通信代价。而直接将通信代价作为目标函数权重之一的启发式遗传算法虽然在一定程度上能够优化通信代价,但依赖于权值的选择和启发式算法自身的收敛效果,优化能力有限。另外,本文算法基于簇进行调度域选择,能有效降低后续启发式算法的运行复杂度。如图7(b)所示,相比直接在站点数据中心全局调度域下进行启发式算法的求解,本文算法对调度域的选择能有效地降低算法的复杂度和运行时间。

然而,本文算法仍存在一定的局限性,当数据中心内剩余的可分配资源十分稀缺时,基于簇代价和利用率的调度域选取算法选取的调度域会扩散到整个数据中心,此时通信代价的优化效果会减弱。

结束语 本文基于未来网络试验设施的资源调度和试验服务需求,设计了集中与分布相结合的架构和分级调度流程,保证了资源管理系统的可扩展和灵活性。中心资源调度管理系统与分布式的节点资源调度管理系统协调合作,实现了全网试验资源的调度管理。同时,针对未来网络试验设施的需求和特点,设计了多目标优化的节点资源调度算法,该算法综合考虑了虚拟机间的通信代价、数据中心资源利用率和资源均衡,实现了站点数据中心内虚拟机资源的调度分配。最后在CloudSim平台进行仿真实验,结果表明,本文算法能够有效地降低站点内虚拟机间的通信代价,同时提高物理机的平均资源利用率和资源均衡。下一步工作将针对算法存在的局限性进行进一步优化;另外,随着未来网络试验设施的建设发展,试验服务的种类和需求会更加丰富,对资源调度算法的要求会不断扩展,考虑不同性能和优化目标的调度算法也将成为未来继续研究的方向。

参考文献

- [1] Wikipedia contributors. Global Environment for Network Innovations[G/OL]. (2019-02-19) [2019-03-01]. https://en.wikipedia.org/wiki/Global_Environment_for_Network_Innovations.
- [2] GENI. Geni-concepts[EB/OL]. [2019-03-01]. <https://www.geni.net/documentation/geni-concepts/>.
- [3] JIANG H, XIAO Y L. Research on Unified Resource Management and Scheduling System in Cloud Environment[J]. *Wireless Personal Communications*, 2018, 102(2): 963-973.
- [4] SINGH S, CHANA I. A Survey on Resource Scheduling in Cloud Computing: Issues and Challenges [J]. *Journal of Grid Computing*, 2016, 14(2): 217-264.
- [5] ESWARAPRASAD R, RAJA L. A Review of Virtual Machine (VM) Resource Scheduling Algorithms in Cloud Computing Environment[J]. *Journal of Statistics and Management Systems*, 2017, 20(4): 703-711.
- [6] NOSRATI M, CHALECHALE A, KARIMI R. Latency Optimization for Resource Allocation in Cloud Computing System [C]// *Computational Science and Its Applications (ICCSA 2015)*. Springer International Publishing, 2015: 355-366.
- [7] MASDARI M, NABAVI S S, AHMADI V. An Overview of Virtual Machine Placement Schemes in Cloud Computing [J]. *Journal of Network and Computer Applications*, 2016, 66 (C): 106-127.
- [8] ALICHERY M, LAKSHMAN T V. Network Aware Resource Allocation in Distributed Clouds[C]// *2012 Proceedings IEEE INFOCOM*. Orlando, Florida, USA: IEEE, 2012: 963-971.
- [9] SHETTY S, YUCHI X B, SONG M. Optimizing Network-aware Resource Allocation in Cloud Data Centers [M]. *Moving Target Defense for Distributed Systems*. Springer International Publishing, 2016: 43-55.
- [10] CANALI C, LANCELLOTTI R, SHOJAFAR M. A Computation-and Network-aware Energy Optimization Model for Virtual Machines Allocation[C]// *Proceedings of the 7th International Conference on Cloud Computing and Services Science - Volume 1: CLOSER*. Porto, Portugal: SCITEPRESS, 2017: 71-81.
- [11] COHEN R, LEWIN EYTAN L, NAOR J, et al. Almost Optimal Virtual Machine Placement for Traffic Intense Data Centers[C]// *2013 Proceedings IEEE INFOCOM*. Turin, Italy: IEEE, 2013: 355-359.
- [12] TIAN W H, XU M X, CHEN Y, et al. Prepartition: A New Paradigm for the Load Balance of Virtual Machine Reservations in Data Centers [C] // *2014 IEEE International Conference on Communications*. Sydney, NSW, Australia: IEEE, 2014: 4017-4022.
- [13] NAGPURE M B, DAHIWALE P, MARBATE P. An Efficient Dynamic Resource Allocation Strategy for VM Environment in Cloud[C]// *2015 International Conference on Pervasive Computing*. St. Louis, Missouri, USA: IEEE, 2015.
- [14] ZHANG M H, REN H L, XIA C H. A Dynamic Placement Policy of Virtual Machine Based on MOGA in Cloud Environment [C]// *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications*. Guangzhou, China: IEEE, 2017: 885-891.
- [15] FAN Z Q, SHEN H, WU Y B, et al. Simulated-annealing Load Balancing for Resource Allocation in Cloud Environments[C]// *International Conference on Parallel & Distributed Computing*. Taipei, Taiwan: IEEE, 2014: 1-6.
- [16] 田文洪, 赵勇. 云计算: 资源调度管理[M]. 北京: 国防工业出版社, 2011: 86-88.