

基于粒子群算法和RBF神经网络的云计算资源调度方法研究

赵宏伟¹ 李圣普²

(沈阳大学信息工程学院 沈阳 110044)¹ (平顶山学院计算机科学与技术学院 平顶山 467000)²

摘要 为了获得云计算资源调度的多目标优化方案,提出了一种云计算资源的动态调度管理框架;然后给出了本系统的基本架构形式,并对其进行了详细设计;其次,建立了以提高应用性能、保证云应用的服务质量和提高资源利用率为目标的多目标优化模型,并结合最新的RBF神经网络和改进粒子群算法对其求解;最后,在CloudSim平台进行了仿真,实验结果表明提出的框架及算法能有效减少虚拟机迁移次数和物理结点的使用数量,在提高资源利用率的同时,能保证云应用的服务质量。

关键词 云计算,神经网络,资源调度,粒子群

中图分类号 TP39 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.3.023

Research on Resources Scheduling Method in Cloud Computing Based on PSO and RBF Neural Network

ZHAO Hong-wei¹ LI Sheng-pu²

(School of Information Engineering, Shenyang University, Shenyang 110044, China)¹

(School of Computer Science and Technology, Pingdingshan University, Pingdingshan 467000, China)²

Abstract In order to implement the multi-objective optimization scheme in cloud computing system, firstly, a dynamic management framework was proposed, providing the structure of the resources scheduling in cloud computing system. Secondly, a multi-objective optimization model was established, which ensures the quality of cloud applications and improves the utilization rate of resources. The RBF neural network and improved particle swarm algorithm were combined to solve the model. Finally, the result of the experiment on the CloudSim simulation platform indicates that the framework and the proposed algorithm can effectively reduce the number of virtual machine migration and the number of used physical nodes, and the scheduling system can not only improve the utilization rate of resources, but also ensure the QoS of cloud application.

Keywords Cloud computing, Neural network, Resource scheduling, Particle swarm

1 引言

云计算是一种计算方式,用户不需要了解云资源的技术基础架构。云计算系统可以分为公共云和私有云两类。公共云由第三方运行和维护,如Google, Amazon等,通过互联网为用户提供资源。而私有云则是由企业自己搭建的,规模一般比较小,但提供了更适合企业运营的IT资源^[1]。

在云计算系统中,由于云资源呈现动态变化的特点,如何进行资源的调度成为了云计算系统中的核心机制。早期的云资源调度研究主要集中在保证应用性能的前提下改进资源利用率^[2]。而当前的研究热点主要是从云计算中心的节能方面进行改进,主要是通过虚拟机的迁移来关闭不必要的服务器,从而降低系统能耗,但迁移会花费一定的时间,导致应用性能的下降。因此,对于云计算中心,一方面为了正确调整和利用硬件资源需要提高资源利用率,另一方面需要保证云应用的服务质量并提高应用性能。然而,多个目标可能存在相互冲

突。因此,为了获得多方面的综合优化,本文设计了云计算中心资源调度框架,综合考虑了用户的服务质量(SLA)、应用性能和资源需求、负载平衡、结点个数、虚拟机性能等多个因素,并提出了云计算资源调度的多目标优化模型与算法,算法基于RBF神经网络和粒子群的仿生智能算法。实验结果表明:与传统的启发式和单目标优化算法相比,本文提出的云计算资源调度框架和算法可有效实现多个目标的优化。

2 相关工作

由于云计算的虚拟化特性,资源可分为物理资源和虚拟资源。因此,云计算环境下的资源调度可分为虚拟资源层的一级调度和物理资源层的二级调度。一级资源调度的含义是用户提交的作业将被切分成若干个任务,调度的核心就是将合适的虚拟资源分配给用户的任务;二级调度的含义是指虚拟资源和物理资源之间的映射关系。云计算的两级调度模型^[3,4]如图1所示。

到稿日期:2015-01-23 返修日期:2015-04-21 本文受辽宁省自然科学基金:基于生物行为的云计算资源调度方法研究(2013020011),辽宁省社会科学基金(L14ASH001)资助。

赵宏伟(1976-),男,硕士,副教授,主要研究方向为大数据、云计算,E-mail:zhw30@163.com;李圣普(1983-),男,硕士,讲师,主要研究方向为无线传感器网络及应用。

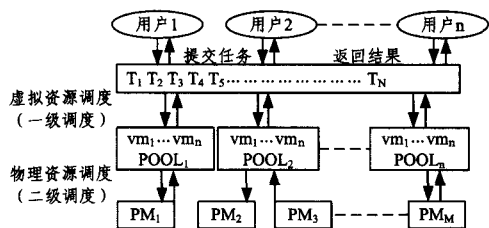


图1 云计算的两级调度模型

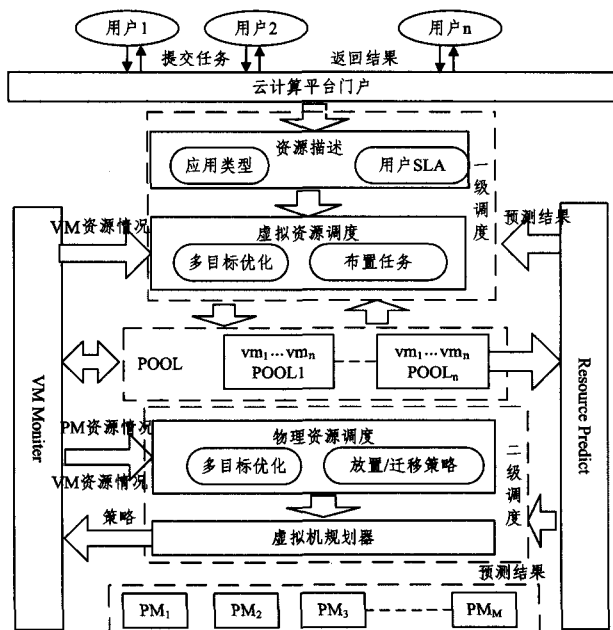


图2 云计算资源的动态调度管理框架

目前,学术界在云计算环境下的资源调度方面已经进行了大量研究工作,文献[5]提出了一种云资源调度模型——云效用最大化模型。与传统调度模型相比,其目标函数不再是最小化最大完工时间,而是以达到效用最大为调度目标,但该算法较简单,难以满足日益增长的云计算规模^[5]。文献[6]根据云计算的弹性化和虚拟化等新特性,提出了云计算中虚拟机放置的自适应管理模型以及虚拟机放置多目标优化遗传算法,但是该云计算系统模型考虑不完美,最优化达不到期望的效果^[6]。国内田冠华等提出了动态的资源调度模型,但该模型仅基于失效规则一方面进行优化,以提高云计算资源调度的效率^[7]。另外,随着仿生智能算法的发展,不少学者提出了基于蚁群算法、粒子群算法、遗传算法等的云计算资源调度模型,这些仿生智能算法具有搜索能力强、并行性等特点,可以快速找到云计算资源最优调度方案,有效地提高了云计算资源的利用效率^[8,9]。然而,仿生智能算法在实际应用中均存在易陷入局部最优解、收敛速度慢等缺陷,有时也不能获得最优资源调度方案。为此,需要对仿生智能算法^[10]进行改进和完善,使云计算资源利用率更高。

另外,多目标优化也吸引了大量学者的注意,有的研究者一般采用线性加权进行求解。这类方法简单方便,但不能得到最优解。还有些研究者采用层次分析法、专家打分法等。这些方法大多缺乏科学性。文献[11,12]都提出了各自的云计算资源调度方法,但这些方法在处理虚拟机放置到结点的优化问题时,都没有考虑虚拟机迁移的开销,并且只考虑了静态的虚拟机放置,而没有涉及到基于虚拟机迁移的动态放置^[11,12]。

可见,云计算中资源调度的研究工作大多只涉及某一方面的优化,或者是考虑应用的服务质量,或者是考虑使用结点和迁移数量最少来提高资源利用率等,但是这些目标的优化是相互冲突的。当前,还有研究将多目标优化问题转化为若干个单目标优化问题来分阶段解决,但大多数只能获得局部的而不是全局的优化解。因此,本文综合考虑 SLA、应用性能、负载均衡、结点使用数量和虚拟机迁移次数等来实现资源调度的多目标优化。

为了获得云计算资源调度的多目标优化方案,本文提出了一种云计算资源的动态调度管理框架,然后给出了本系统的基本架构形式,并对其进行了详细设计;最后提出了一种基于 RBF 神经网络和改进的 PSO 的多目标资源调度方法,其在提高资源利用率的同时,保证了云应用的服务质量。

3 云计算资源的动态调度管理框架

本文根据云计算资源调度系统的特点,提出了一种综合考虑用户的服务质量(SLA)、应用性能和资源需求、负载均衡、结点个数、虚拟机性能等多个因素的云计算调度系统框架,调度系统的结构如图2所示。

云计算资源调度系统分为虚拟资源调度和物理资源调度^[13,14]。

定义1(虚拟资源调度) 主要作用是为用户提交的任务提供相应的虚拟机资源,建立起资源与任务之间的有效映射关系,以达到虚拟机资源的有效使用。

定义2(物理资源调度) 主要功能是把虚拟资源合理布置到物理机资源上,以达到物理机资源的最优配置。

在此云资源的调度框架中,一方面,一级调度的云计算门户模块负责接收各个用户提交的任务,将任务提交给资源描述模块,判断任务调用的应用的类型和任务所要求的 SLA,然后由资源预测模块(Resource Predict)进行资源调度的预测,预测结果发布给虚拟资源调度模块,虚拟资源调度模块根据预测的结果和虚拟机(VM)资源的使用情况,采用多目标优化方法进行资源配置和任务布置。首先,云计算系统中的所有资源请求都会经过资源描述和资源预测模块处理,而虚拟资源调度模块根据资源预测结果和 VM 资源使用情况进行资源的调度,将其分配给各个资源池。其次,为保证资源使用的负载均衡并提高资源的利用率,系统设计了不同应用类型的资源池,包括计算密集型、存储密集型和应用密集型等类的资源池,调度系统根据用户提交任务的情况进行不同资源池的调度。这时,资源池也将把本资源池的情况发布给资源监视模块(VMMonitor)。最后,把资源请求和任务处理的结果返给用户或应用。另一方面,二级调度的物理资源调度模块根据 VM 资源情况和物理机资源(PM)情况,也采用多目标优化的方法产生资源的放置或迁移策略,最后由虚拟机规划器发送给 VMMonitor 进行资源放置和迁移。这样,整个调度系统充分考虑了用户、应用类型和资源本身多个优化目标的特点,并采用多目标优化的方法,不但提高了云计算的调度效率和资源利用率,也避免了资源调度成为云计算系统的瓶颈。

本文第一级调度是从已建立的虚拟机模板中选择,即建立虚拟机与任务之间的一一映射。本文重点研究第二级调度的方法,以提高应用性能和保证云应用的服务质量、提高资源利用率为目标,寻找合适的物理资源,完成虚拟机到物理主机间的映射。

4 云物理资源调度模型和多目标优化算法

在云计算环境中,资源调度分为两级:一级是虚拟资源调度,主要是为用户任务分配可用的虚拟资源;另一级是物理资源调度,主要是虚拟机布置到适合的物理机资源的调度。

4.1 云计算物理资源调度模型

目前的大量研究表明,云计算资源调度是一个 NP 难题,如果采用传统穷举算法对所有可能的云算资源调度问题进行求解,该算法的时间复杂度为 $O(n^m)$ 。本文假设云计算环境中的物理资源向量为 $P(p_1, p_2, \dots, p_k)$,需要放置的虚拟资源向量为 $V(v_1, v_2, \dots, v_n)$,布置在虚拟机上的应用向量为 $H_i(h_{i1}^1, h_{i2}^2, \dots, h_{in}^n)$,应用被布置在虚拟机上,虚拟机被放置在物理资源 P 上,得到的位向量为 $H_i(h_{i1}^1, h_{i2}^2, \dots, h_{in}^n)$, $h_{ij}^k=1$ 表示应用 a_k 布置在虚拟资源上,而虚拟资源放置在物理资源上。为了获得虚拟机资源所需要的最少物理资源的最优放置,给出了如下目标函数:

$$\min \sum_j p_j, \min \sum_i m_i \text{ 和 } f(z) \quad (1)$$

其中, p_j 表示物理结点的使用数量最少, m_i 表示虚拟机迁移次数最少, $f(z)$ 设定为惩罚函数。物理资源与虚拟资源的约束关系可以通过式(2)~式(4)表示,其中 R_c^a 则表示每个应用在虚拟资源的 CPU 需求, P_c 表示每个物理资源的 CPU 大小; R_m^a 表示每个应用在虚拟资源的内存需求, P_m 表示每个物理资源的内存大小。由此可得约束:

$$R_c^a \cdot H_i < P_c(p_i) \quad \forall a \in APP \quad \forall p_i \in P \quad (2)$$

$$R_m^a \cdot H_i < P_m(p_i) \quad \forall p_i \in P \quad (3)$$

$$PI_a \geq SLO_a \quad \forall p_i \quad (4)$$

PI_a 表示应用性能指标, SLO_a 表示应用服务级目标。

定义 3(应用服务级目标/SLO) 其指布置在云计算上的应用服务在提供服务时应该达到的最低要求。

定义 4(应用服务满意度) 在一个时间跨度内,布置在云计算上的应用的 PI_a 能满足约束(4)的个数与总共应用个数之比。

在每个物理资源调度中,通过图 2 的资源池(PPOOL)模块判断应用的 PI_a 是否能满足约束(4),以此预判需要对哪些虚拟机进行迁移,并监测云计算系统的应用服务满意度,当应用服务满意度过低时,启动虚拟机放置的物理资源调度;同时可以预测一个物理结点的处理能力,最终为物理资源调度中虚拟机放置和迁移策略的制定提供支持。物理资源调度中虚拟机放置的优化目标是在保证多个应用的服务级目标的情况下,使物理结点的使用数量和虚拟机迁移次数最少。在建立的虚拟机资源布置的约束满足模型的基础上,物理资源调度模型通过求解约束满足模型获得优化的虚拟机资源放置方案,然后基于优化的虚拟机资源放置方案设计和实现从物理资源到虚拟资源的分配算法,进而实现云计算资源的优化分配。

4.2 基于粒子群(PSO)的多目标优化算法

4.2.1 粒子群优化算法

粒子群优化算法(PSO)是模拟鸟群、鱼群,通过个体间的协作能够记忆个体最优和全局最优信息。种群中的每一个粒子即为代表待优化目标问题的一个可能解。设在一个 D 维的搜索空间中,由 m 个粒子组成的种群 $x = (x_1, x_2, x_3, \dots, x_m)$,其中第 i 个粒子位置为 $x_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{iD})$,粒子的速度为 $v_i = (v_{i1}, v_{i2}, v_{i3}, \dots, v_{iD})$, c_1, c_2 是学习因子; R_1, R_2 是均匀分布在 $(0,1)$ 区间的随机数。 p_{id} 是第 t 时刻粒子 i 当

前的最优位置, p_{gd} 是第 t 时刻粒子群的全局最优位置。则粒子 x_i 的速度和位置更新公式分别为:

$$v_{id}(t) = \chi(v_{id}(t-1) + R_1 c_1 (p_{id} - x_{id}(t-1)) + R_2 c_2 (p_{gd} - x_{id}(t-1))) \quad (5)$$

$$x_{id}(t) = x_{id}(t-1) + v_{id}(t) \quad (6)$$

$$\chi = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|} \quad (7)$$

其中, $\varphi = c_1 + c_2$, $\varphi > 4$ 。在 t 时刻, m 是粒子群内粒子的个数; x_{id} 是粒子当前位置; v_{id} 表示粒子 x_i 的当前速度。如果粒子的邻域包含了整个粒子群,则以上的算法即为全局模式的粒子群算法 global-PSO,否则即为局部模式的粒子群算法 local-PSO。

4.2.2 基于 PSO 的 RBF 神经网络的优化过程

本节利用 PSO 算法操作简单、容易实现等特点,对 RBF 神经网络的参数和连接权重进行优化,并基于 PSO 优化后的 RBF 神经网络进行云计算应用资源需求的预测^[15]。PSO 优化 RBF 神经网络可以表述如下:

首先,设定从 PSO 粒子的维度空间到 RBF 神经网络的映射的参数。PSO 算法中每个粒子的维度都对应 RBF 神经网络中的每一个参数值。也就是 RBF 神经网络中每个参数(包括中心点 c_i 、宽度系数 δ_i 和权值 w_{ij}),都对应着 PSO 算法中某个粒子的一个维度。

其次,设定 RBF 均方误差为 PSO 算法的适应度函数。这里将串码映射为所表示的个体的同时,也需要对网络中的参数进行优化。优化训练得到了均方误差最小的最佳连接权值。

最后,将 RBF 连接权值的参数编码为实数向量,并表示为种群中的个体,进行迭代。把算法迭代过程中产生的个体向量还原为 RBF 的权值,并根据设定参数计算出 RBF 产生的均方误差。当误差精度达到指定的临界值时训练停止,否则迭代继续,直到到达规定的迭代最大次数时停止。

为了验证 PSO-RBF 算法的有效性,采用 MATLAB7 软件中的神经网络工具箱进行 PSO-RBF 神经网络的仿真实验。这里采集云计算平台的不同应用需求量,共 150 个数据,其中 100 作为训练数据,反复训练 30 次,用训练好的 PSO-RBF 神经网络对后 50 个数据进行样本测试。图 3、图 4 横轴表示迭代次数,纵轴表示误差。图 3 中,当迭代到第 220 次时达到预设精度;图 4 中,当迭代到第 140 次时达到预设精度。可以看出,PSO-RBF 算法在更短时间内可以把误差减到最小,收敛效果更好。图 5 展现的是训练后的 RBF 网络在 50 个测试样本上的逼近能力,实线为目标值,虚线为网络的实际输出值,实验仿真结果表明了该模型的有效性,其可以为输入实测数据进行应用需求量预测。

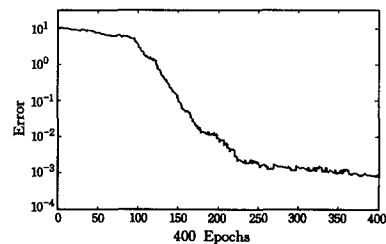


图 3 RBF 神经网络收敛曲线

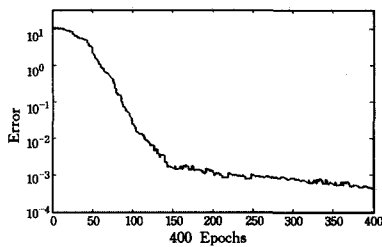


图4 PSO优化的RBF神经网络收敛曲线

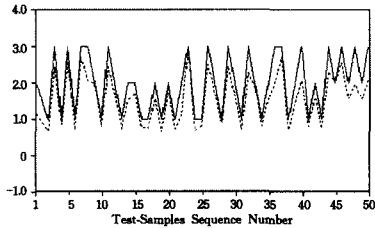


图5 网络输出与目标值对比

4.2.3 PS²O粒子群优化算法及其算法复杂度分析

粒子群优化算法 PS²O 主要从动态多群体协作进行改进,每次迭代都增加子群间的最优的个体信息。动态多群体协作提高了算法的收敛速度,并引入最大速度 $v_d^{\max}$ 来选择产生变异粒子,通过变异粒子的飞行克服陷入局部最优的缺点,保持了群体的多样性,以此来提高算法的性能和最优解的精度。 $P=\{S_1, S_2, \dots, S_M\}$ 表示共 M 个子群,每个子群 $S_k=\{X_1^k, X_2^k, \dots, X_N^k\}$ 有 N 个个体,则粒子 x_i 的速度和位置更新公式分别为:

$$v_{id}^k(t) = \chi(v_{id}^k(t-1) + R_1 c_1 (p_{id}^k - x_{id}^k(t-1)) + R_2 c_2 (p_{gd}^k - x_{id}^k(t-1)) + R_3 c_3 (p_{gd}^0 - x_{id}^k(t-1))) \quad (8)$$

$$x_{id}^k(t) = x_{id}^k(t-1) + v_{id}^k(t) \quad (9)$$

$$v_{id}^k = \begin{cases} v_{id}^{\max}, & v_{id}^k > v_{id}^{\max} \\ -v_{id}^{\max}, & v_{id}^k < -v_{id}^{\max} \end{cases} \quad (10)$$

其中, x_{id}^k 表示第 k 个子群中第 i 个粒子的位置, p_{id}^k 是个体最优位置信息, i 最优位置信息是 x_{id}^k , p_{gd}^k 是在子群 k 中的最优位置, p_{gd}^0 是在第 k 子群的邻居子群最优位置。 c_1, c_2, c_3 是学习因子; R_1, R_2, R_3 是均匀分布在 $(0, 1)$ 区间的随机数, $\varphi = c_1 + c_2 + c_3, \varphi > 4$ 。

改进 PSO 算法的伪代码如下:

```
Set t := 0;
INITIALIZE. Randomize n swarms
each possesses m particles;
WHILE (the termination conditions are not met)
//是否满足结束条件
FOR (each swarm k) //子群遍历
     $v_{id}^k = v_d^{\max} \cdot v_{id}^k = -v_d^{\max}$ 
    Find in the  $k^{\text{th}}$  swarm neighborhood, the point with the best fitness;
    Set this point as  $p_{gd}^k$ ;
//找到第 k 子群的邻居子群的最优位置
    FOR (each particle i of swarm k)
//子群粒子遍历
        Find in the particle neighborhood, the point with the best fitness;
        Set this point as  $p_{id}^k$ ;
//找到第 k 子群的最优位置
    Update particle velocity using equations (8);
```

Update particle position using equations (9);

//产生变异粒子

END FOR

END FOR

Set t := t + 1;

END WHILE

另外,本节选择了 Sphere、Rosenbrock、Griewank 这 3 个常用的标准测试函数^[16]用于测试提出的 PS²O 算法的性能,并与 PSO 算法进行了性能比较。3 个函数中前两个是单峰函数,后一个是多峰函数。实验工具采用 MATLAB7,种群的初始设置如表 1 所列。

表1 测试函数的参数设置

Test Function	Dim.	Population Size	Initial Hypercube
Sphere	20	30	$[50, 100]^n$
Rosenbrock	20	30	$[15, 30]^n$
Griewank	20	30	$[300, 600]^n$

对于这 3 个函数, $v_{\max}$ ($v_{\min} = -v_{\max}$) 的值分别设置为 100, 100, 600。对 PSO 和 PS²O 两种算法中的公有参数: $c_1 = c_2 = 2$, 惯性权重的大小设置为从 0.8 线性递减至 0.3。最大速度设置为搜索空间大小的一半。每个算法独立执行 50 次。

实验的结果如表 2 所列。从表中可以看出,在 3 种测试函数上, PS²O 算法获得的平均适应度值几乎都要好于 PSO 算法,在 Sphere、Rosenbrock 2 个函数上的表现尤为明显。此外,在 50 次独立实验中,虽然在时间复杂度 $T(n)$ 上 PS²O 要略逊于 PSO 算法,但是 PS²O 算法获得的标准方差也要小于 PSO 算法,并且 PS²O 算法能够避免收敛较快而陷入局部最优,因此 PS²O 比 PSO 算法更稳定,表明其具有较强的鲁棒性。

表2 PS²O 与 PSO 结果的比较

Algorithm	Mean	StdDev	T(n)
Sphere			
PSO	1.8809e-07	1.8607e-07	$O(n)$
PS ² O	1.0232e-045	1.0725e-045	$O(n \log_2 n)$
Rosenbrock			
PSO	3.35891e+03	7.1166e+003	$O(n)$
PS ² O	2.7328e+01	3.4790e-001	$O(n \log_2 n)$
Griewank			
PSO	2.2178e-02	4.0754e-02	$O(n)$
PS ² O	1.7467e-05	9.7738e-05	$O(n \log_2 n)$

4.3 基于 PSO-RBF 和 PS²O 多目标资源调度优化

采用 PSO-RBF 预测结果和改进的 PS²O 多目标优化的资源分配算法,求解云计算的物理资源调度优化问题,每个粒子代表一个物理资源分配方案。对于整个云计算中心,考虑到不同应用类型对资源的需求不同(如科学计算应用需要强大的计算资源,多媒体应用需强大的数据存储能力),为了达到各个资源子群的相互协作,使互补的多种应用资源需求可以部署到单个物理主机上,把整个资源池划分为 M 个资源子群,每个子群有 N 个粒子。

定义 5(资源子群) 根据不同应用类型对资源需求的不同,将应用放置在相应类型的虚拟机,最终形成各类资源子群。

在物理资源调度过程中,首先通过 PSO-RBF 得到 RBF 神经网络的最优参数,然后将已有应用资源需求量数据分成两部分,一部分数据作为训练样本,另一部分作为实测数据。首先,用训练数据针对 RBF 神经网络进行训练;之后,用训练

好的 PSO-RBF 神经网络预测应用资源需求量,并将预测结果发送给 PS^2O 作为粒子适应度计算选择的依据。

本文采用均匀方法产生初始化粒子群,并选择文献[6]中的适应度函数的设计方案,执行式(8)和式(9),产生下一代粒子的搜索速度、当前位置、最优位置和全局最优位置;然后基于最大速度 v_{max} 来选择产生变异粒子;最后经过多次迭代,求出最优解。优化流程如图 6 所示。

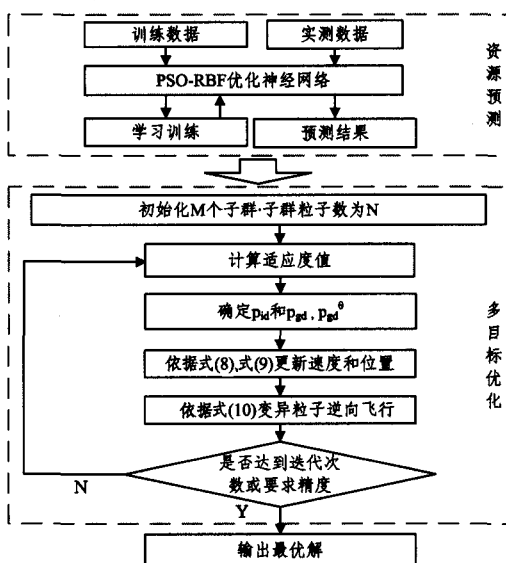


图 6 基于 PSO-RBF 和 PS^2O 的优化流程

5 实验及结果分析

为了验证该 PSO-RBF 和 PS^2O 算法在云计算资源调度策略中的可行性,本文采用 CloudSim 针对云计算环境中的调度策略进行模拟仿真测试。本文对平台进行了扩展,重写一系列的类,模拟了云计算数据中心,编程工具为 Myeclipse9.0。本实验共分两组进行,第一组实验统一采用 PSO-RBF 神经网络并预测应用资源需求量,用基本粒子群算法(PSO)作为对比模型来衡量 PS^2O 的资源调度方案的优劣;第二组实验主要是把采用 PSO-RBF 进行资源预测的 PS^2O 算法和没有资源预测的 PS^2O 算法进行对比,以验证应用资源预测的效果。

第一组实验,在云计算中心设置生成 30 个物理资源,每个结点装备 3.0GHz 的 CPU,2GB 内存,然后对它们进行虚拟化,把虚拟资源划分为 3 个子群,最大迭代次数为 100。学习因子 $c_1=c_2=c_3=1.333$ 。两种物理资源调度算法 PS^2O 和 PSO 均迭代 100 次,并设置用户提交 100 个任务,分别进行 30 次实验,包括对处理器、内存物理资源的使用情况和可用资源的负载均衡、平稳性进行实验(假设一级调度的算法相同)。图 7、图 8 对比了 PS^2O 算法和 PSO 算法对 CPU 资源调度和内存资源调度的利用率,可以看出在资源数一定,任务数逐渐增加的前提下,两种算法的资源利用率均会上升,但因为 PSO 算法使用的物理资源数多,使其任务请求更分散到各个节点中。图 9 比较了两种算法的可用资源负载均衡度,可以看出本文设计的算法在任务数量增加的情况下,其资源负载较平稳,均衡度较好。图 10 比较了两种算法完成相同数量的任务所使用的时间分布情况,可以看出 PS^2O 所需的时间波动范围相对较小,执行时间也较少。

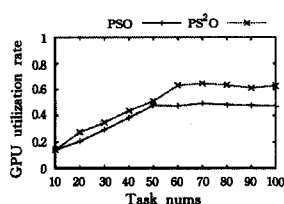


图 7 两种算法 CPU 资源利用率的比较

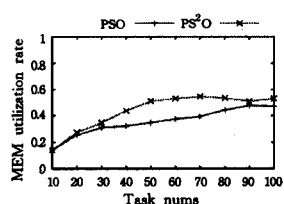


图 8 两种算法内存利用率的比较

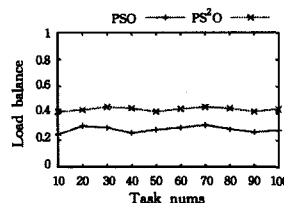


图 9 两种算法负载均衡度的比较

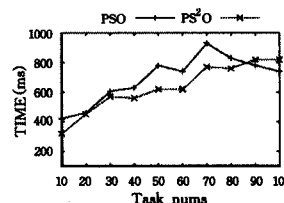


图 10 两种算法执行时间的比较

因此,采用 PS^2O 算法比 PSO 算法在 CPU 和内存上的资源利用效率更高,而且随着任务数的增加, PS^2O 算法的波动比 PSO 算法的波动小,显示出 PS^2O 算法的平稳特性,其更适合在云计算数据环境下资源调度。

为了验证应用资源预测的重要性,最后进行一组实验,主要是把基于 PSO-RBF 进行资源预测的 PS^2O 和 PSO 算法和没有资源预测的 PS^2O 算法进行对比。如图 11 所示,PSO-RBF+ PS^2O 算法相对平稳,用 30 次迭代基本达到最优,具有较好的收敛效果。 PS^2O 算法达到基本最优的迭代次数为 90,与 PSO-RBF+PSO 算法达到基本最优的迭代次数相差不多。这说明,基于 PSO-RBF 应用的资源预测起到了优化的效果,PSO-RBF+ PS^2O 算法达到基本最优的迭代次数要小于没有资源预测的调度算法 PS^2O 。

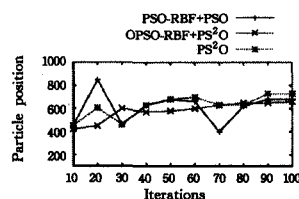


图 11 PSO-RBF+ PS^2O 、PSO 和 PS^2O 算法收敛的比较

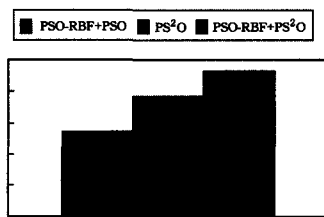


图 12 3 种算法的服务目标满足率

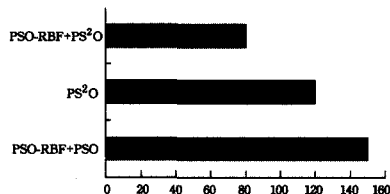


图 13 3 种算法的虚拟机迁移次数对比

在第一组实验的条件下,图 12、图 13 表示 PSO-RBF+ PS^2O 算法在满足服务级目标约束条件下,应用的服务目标满

(下转第 150 页)

- ber Physical Systems; Network Topology, Interdependence and Cascading Failures [J]. IEEE Transactions on Parallel & Distributed Systems, 2015, 26: 2340-2351
- [2] Jakobson G. Mission Cyber security situation assessment using impact dependency graphs[C]// 2011 Proceedings of the 14th International Conference on Information Fusion (FUSION). IEEE, 2011: 1-8
- [3] Barnett A, Smith S R, Whittington R P. Using Causal Models to Manage the Cyber Threat to C2 Agility: Working with the Benefit of Hindsight[R]. Defence Science and Technology Lab Porton Down(United Kingdom), 2014
- [4] Neuman C, Tan K. Mediating Cyber and Physical Threat Propagation in Secure Smart Grid Architectures[C]// 2011 IEEE In-

- ternational Conference on IEEE Smart Grid Communications (Smart Grid Comm). 2011: 238-243
- [5] Cheng Yi, Deng Ju-lia, Li J, et al. Cyber Defense and Situational Awareness[M]. Springer International Publishing, 2014
- [6] Halappanavar M, Choudhury S, Hogan E, et al. Towards a network-of-networks framework for cyber security [C] // 2013 IEEE International Conference on IEEE Intelligence and Security Informatics (ISI). 2013: 106-108
- [7] Berge C. Graphes et hypergraphes [M]. Dunod, Paris-Brussels-Montreal, Que., 1970
- [8] Wang Zhong-tuo. Introduction to System Engineering [M]. Beijing: Electronic Industry Press, 2012 (in Chinese)
- 王众托. 系统工程引论[M]. 北京: 电子工业出版社, 2012

(上接第 117 页)

足率高于 PS^2O 算法和 $PSO-RBF+PSO$ 算法, 虚拟机迁移次数少于 PS^2O 算法和 $PSO-RBF+PSO$ 算法。

总之, $PSO-RBF+PS^2O$ 算法在迭代次数较多、节点较多、有效资源较少的情况下工作效率明显比 PSO 资源分配算法效率高。本算法在提高了云计算中心资源利用率的同时, 也保证了云应用的服务质量, 同时具有较高的应用性能。

结束语 本文首先设计了云计算资源调度的两级动态调度管理框架, 并给出了框架系统的具体结构和形式; 其次, 采用 $PSO-RBF$ 算法对应用的资源需求量进行预测, 并提出了一种多目标优化算法 $PSO-RBF+PS^2O$, 该算法综合考虑了云应用的服务质量、应用性能和资源利用率; 最后, 在 Cloud-Sim 平台进行了仿真, 结果表明此调度模型和多目标算法能有效地提高云计算系统中应用的服务质量和资源的利用率。下一步将考虑在开源云计算平台如 OpenStack 上应用文中的调度策略, 并验证相关策略的有效性。

参 考 文 献

- [1] Armbrust M, Fox A, Griffith R. A view of Cloud computing[J]. Communications of the ACM, 2010, 53(4): 50-58
- [2] Rivoire S, Shan M, Ranganathan P. A balanced energy-efficiency benchmark[C]// ACM SIGMOD International Conference on Management of Data. Beijing, China, 2007: 376-380
- [3] Goiri I, Guitart J, Torres J. Characterizing Cloud Federation for Enhancing Providers Prot[C]// 3rd International Conference on Cloud Computing. Miami, USA, 2010: 123-130
- [4] Goiri I, Juli A F, Torres J. Energy-aware Scheduling in Virtualized Data Centers[C]// 12th IEEE International Conference on Cluster Computing. Heraklion, Greece, 2010: 58-65
- [5] Shi X L, Xu K. Utility Maximization Model of Virtual Machine Scheduling in Cloud Environment [J]. Chinese Journal of Computers, 2013, 36(2): 252-262 (in Chinese)
- 师雪霖, 徐格. 云虚拟机资源分配的效用最大化模型[J]. 计算机学报, 2013, 36(2): 252-262
- [6] Li Q, Hao Q F, Xiao L M. Adaptive Management and Multi-Objective Optimization for Virtual Machine Placement in Cloud Computing[J]. Chinese Journal of Computers, 2011, 34(12): 2256-2257 (in Chinese)
- 李强, 郝沁汾, 肖利民. 云计算中虚拟机放置的自适应管理与多目标优化[J]. 计算机学报, 2011, 34(12): 2256-2257
- [7] Tian G H, Ment D, Zhan J F. Reliable Resource Provision Policy for Cloud Computing[J]. Chinese Journal of Computers, 2010,

- 33(10): 1859-1872 (in Chinese)
- 田冠华, 孟丹, 詹剑锋. 云计算环境下基于失效规则的资源动态提供策略[J]. 计算机学报, 2010, 33(10): 1859-1872
- [8] Lei D M, Yan X P. Intelligent Multi-objective optimization algorithm and Application[M]. Beijing: Science Press, 2009 (in Chinese)
- 雷德明, 严新平. 多目标智能优化算法及其应用[M]. 北京: 科学出版社, 2009
- [9] Coello C, Carlos A, Lechuga M S. MOPSO: a proposal for multiple objective particle swarm optimization[C]// Proceedings of IEEE Congress on Evolutionary Computation. Honolulu, USA: IEEE, 2002: 1051-1056
- [10] Xu X B, Zheng K F, Li D. New chaos-particle swarm optimization algorithm[J]. Journal on Communications, 2012, 33(1): 24-30 (in Chinese)
- 胥小波, 郑康峰, 李丹. 新的混沌粒子群优化算法[J]. 通信学报, 2012, 33(1): 24-30
- [11] Wang J Z, Chen Y J. Research on I/O Resource Scheduling Algorithms for Utility Optimization Towards Cloud Storage[J] Journal of Computer Research and Development, 2013, 50(8): 1657-1666 (in Chinese)
- 王健宗, 谌炎俊. 面向云存储的 I/O 资源效用优化调度算法研究[J]. 计算机研究与发展, 2013, 50(8): 1657-1666
- [12] Luo J Z, Jin J H, Song A B. Cloud computing: architecture and key technologies[J]. Journal on Communications, 2011, 32(7): 2-19 (in Chinese)
- 罗军舟, 金嘉晖, 宋爱波. 云计算: 体系架构与关键技术[J]. 通信学报, 2011, 32(7): 2-19
- [13] Grit L, Irwin D, Yumerefendi A. Virtual Machine Hosting for Networked Clusters[C]// Proceedings of the 2nd International Workshop on Virtualization Technology in Distributed Computing. Durham, USA, 2006: 1-5
- [14] Andr H, Lagar-Cavilla S, Whitney J A. SnowFlock: virtual machine cloning as a first-class cloud primitive[J]. ACM Trans Comput Syst, 2011, 29(1): 1-45
- [15] Han H G, Qiao J F, Bo Y C. On Structure Design for RBF Neural Network Based on Information Strength [J]. Acta Automatica Sinica, 2012, 38(7): 1083-1090 (in Chinese)
- 韩红桂, 乔俊飞, 薄迎春. 基于信息强度的 RBF 神经网络结构设计研究 [J]. 自动化学报, 2012, 38(7): 1083-1090
- [16] Abido M A. Optimal Power Flow using Particle Swarm Optimization[J]. International Journal of Electrical Power & Energy Systems, 2002, 24(7): 563-571