

DOI:10.13718/j.cnki.xsxb.2017.07.009

基于自适应拍卖和迁移机制的 云计算资源调度算法^①

王 瑞 娜

三门峡职业技术学院 信息传媒学院, 河南 三门峡 472000

摘要: 为了解决以往云计算环境下资源调度中的不合理问题, 提出一种基于自适应拍卖和迁移机制的云计算资源调度算法. 该算法综合网络带宽和拍卖机制的特点, 首先对竞拍时限内的申请用户进行排序, 筛选出符合用户服务质量和云服务商最低成本的资源调度方案; 接着, 对热点资源需求进行迁移, 实现资源的合理应用. 仿真实验表明该算法能够在满足用户资源需求、降低云服务商运营成本的基础上实现云环境下资源的自适应合理配置.

关 键 词: 云计算; 资源; 拍卖; 迁移

中图分类号: TP182

文献标志码: A

文章编号: 1000-5471(2017)07-0054-09

云计算是基于互联网的相关服务的增加、使用和交付模式, 是分布式计算、虚拟化等传统计算模式与网络技术发展相融合的产物^[1]. 随着云计算应用领域的不断扩展, 云资源的动态合理分配业已成为影响和制约用户服务质量(quality of service, QoS)和云服务商运营成本的重要因素, 如何快速响应用户端需求和动态合理地调整云资源成为解决问题的关键.

在云资源中, 虚拟机(virtual machine, VM)资源是云计算中一类极具代表性的资源, 占有相当大的比重, VM资源的调度和分配不仅关系到 QoS, 而且也直接关系到云服务商的收益. 目前, 针对 VM资源调度问题产生了一系列资源调度算法^[2-20].

一方面, 文献[2]提出了基于静态思想的 VM资源分配方案, 该方案有助于提升云服务商的运营效益, 但无法适应动态的云资源环境; 文献[3]提出的 EAC资源管理模型可以用于处理小批量的资源调配任务; 文献[4]针对文献[2]中的不足, 针对云环境下的资源动态特性, 提出一种 VM动态配置策略; 文献[5]提出可以对用户的需求进行评估从而决定云服务商应当提供多少 VM资源; 文献[6]采用信号与统计学理论对 VM资源需求进行估测; 文献[7]采用微分计算法模拟曲线来估计最优的动态 VM资源集合, 取得了良好的效果, 但计算复杂度较高; 文献[8-11]分别提出了基于拍卖机制的云计算资源分布算法, 取得了显著的效果. 然而, 文献[8]中的算法需要对工作负荷做先验预测, 文献[9]不需要先验预测, 但计算复杂度过高; 文献[10-11]对文献[8-9]的优势进行了很好地结合, 充分考虑了云计算中包括 VM资源、网络带宽等多方面的因素, 对经典的拍卖机制进行了完善, 然而, 没有能够将竞拍时限因素考虑在内, 在一定程度上可能造成 VM资源的浪费.

① 收稿日期: 2016-03-28

基金项目: 河南省高等学校重点科研项目(15B520026).

作者简介: 王瑞娜(1984-), 女, 河南灵宝人, 讲师, 硕士, 主要从事计算机网络应用研究.

另一方面, 文献[12]针对资源的评估和预测进行了描述; 文献[13-16]中涉及的 DRMonitor, Ganglia, NWS 和 MDS 预测系统可以对云环境中的资源进行简单地评估, 得出诸如 CPU 利用率、网络带宽等一系列数据, 然而这些数据指标无法适应云计算环境的动态变化特点; 文献[17]构建了一种资源配置框架, 可以针对申请用户的需求配置相应资源; 文献[18]采用信号处理和统计学方法对资源需求进行动态预测, 然而, 上述文献并未对云服务商的可用资源情况进行掌握和评估。

本文正是在上述背景下展开, 综合考虑网络带宽以及竞拍机制的特点: 一方面对竞拍时限内的申请用户进行排序, 筛选出符合用户服务质量和云服务商最低成本的资源调度方案; 另一方面, 对云服务商的可用 VM 资源进行评估, 及时将热点资源需求进行迁移, 实现资源的合理应用. 仿真实验表明该算法能够在满足用户资源需求、降低云服务商运营成本的基础上实现云环境下资源的自适应合理配置. 文中算法采用 CloudSim 平台进行仿真, 仿真结果表明, 本文算法能够在满足用户资源需求、降低云服务商运营成本的基础上实现云环境下资源的自适应合理配置.

1 云计算基本模型

云计算可以为大量用户同时提供大批量的任务请求, 一旦接收到服务请求, 云服务商便会根据用户的不同请求、用户为对应云计算资源支付的不同费用分配相应的计算资源. 图 1 给出了云计算的基本模型示意图.

云计算资源尤其是 VM 资源性能是云计算服务性能中的重要指标, 能够在相当程度上衡量云服务商向用户提供服务的概率及服务能力的水平. 通常, VM 资源性能评估主要基于对 VM 可用资源信息的掌握, 通过对 VM 资源可用度进行评估实现对当前已用 VM 资源和可用 VM 资源的描述, 以备后期的资源调度、分配或迁移.

为了能够有效地对云计算中的动态变化资源信息进行描述, 本文提出 4 个动态资源信息描述指标: 资源需求量、处理能力、负荷指数、成本指数.

令 VM 资源集合为 $S = \{S_1, S_2, \dots, S_n\}$, n 为 VM 资源的数量.

1) 资源需求量(RRQ): 单位时间内云服务商接收到的服务请求量. RRQ 数值越大, 表明用户对云服务商资源的需求量越大. 若 VM 资源 $S_i (1 \leq i \leq n)$ 单位时间内接收到的服务请求为 $RRQ_i (1 \leq i \leq n)$, 则集合 S 的需求指数为:

$$RRQ = \sum_{i=1}^n RRQ_i \quad (1)$$

2) 处理能力(DA): 单位时间内 VM 资源完成的服务请求量. DA 数值越大, 表明云服务商资源对用户任务请求的处理能力越强. 若 VM 资源 $S_i (1 \leq i \leq N)$ 单位时间内完成的服务请求量为 $DA_i (1 \leq i \leq N)$, 则集合 S 的处理指数为:

$$DA = \sum_{i=1}^n DA_i \quad (2)$$

3) 负荷指数(LI): VM 资源完成一个用户服务的平均时间与用户相邻两次服务请求间隔时间的比值. LI 数值越大, 表明云服务商资源的负荷越大. 若 VM 资源 $S_i (1 \leq i \leq N)$ 完成一个用户服务的平均时间为 $t_i (1 \leq i \leq N)$, 用户相邻两次服务请求间隔时间 t_d , 则负荷指数为:

$$LI = \frac{t_i}{t_d} \quad (3)$$



图 1 云计算基本模型示意图

4) 成本指数(CI): 衡量云服务商维护 VM 资源正常运行的成本. 若 VM 资源 $S_i (1 \leq i \leq N)$ 运行时维护成本为 C_{iR} , 闲置时维护成本为 C_{iI} , 则云服务商资源维护的总成本指数位于下列区间 CI:

$$CI = \left[\sum_{i=1}^n C_{iI}, \sum_{i=1}^n C_{iR} \right] \quad (4)$$

显然, 云服务商要想获得基础收益, 必须保证单位资源对应的用户群竞拍价格 P_i 位于区间 $[C_{iI}, C_{iR}]$. 因为若 $P_i < C_{iR}$, 则云服务商损失为 $C_{iR} - P_i$, 但该损失必须满足 $C_{iR} - P_i < C_{iI}$, 即必须让 VM 资源运行的损失小于其闲置时的损失. 因而, 用户群竞拍价格 P_i 必须满足 $C_{iI} \leq P_i \leq C_{iR}$.

2 基于自适应竞拍机制的 VM 资源调度算法

本文提出的基于自适应竞拍机制的云计算 VM 资源调度算法分为两大部分: 基于网络带宽和拍卖机制的支付算法以及基于迁移机制的资源调配模型.

2.1 基于网络带宽和拍卖机制的支付算法

综合网络带宽以及网络竞拍机制等因素, 可先行构建云服务商和用户群的基本模型:

云服务商 S_{cloud} : 拥有 VM 资源种类数为 K , VM 资源总数量为 n_S , 单位 VM 资源 $S_{k,i} (1 \leq k \leq K, 1 \leq i \leq n_S)$ 的运行维护成本和闲置维护成本分别为 $C_{k,i,R}$, $C_{k,i,I}$, 某时刻公布的 VM 资源竞拍时限为 $[t_{\min}, t_{\max}]$.

用户群 U : 用户群中的用户个体数为 n_U , 单位用户 $U_p (1 \leq p \leq n_U)$ 对云服务商提出的服务请求为 $(S_{k,i,P}, b_p, t_p, P_p)$, 其中 $S_{k,i,P}$ 表示用户 U_p 对第 k 类 VM 资源的需求量为 i , b_p 为用户 U_p 的带宽需求, t_p 为用户 U_p 的竞价时刻, P_p 是用户 U_p 对该服务请求所能承担的最高价格. 需要说明的是: t_p 时刻必须位于云服务商公布的竞拍时限以内, 否则该用户服务请求无效; 此外, 用户的最终支付价格 $P_{p'}$ 满足 $P_{p'} \leq P_p$.

与以往文献中提出的竞拍机制有所不同, 本文提出的算法模型加入了竞拍时限限制, 该条件有助于提升云服务商 VM 资源的利用效率, 云服务商不会不间断地在网络中公布自身资源的竞拍时限, 从而将一些已经提出用户服务请求但却长时间无响应且一直占据带宽资源的用户及时剔除, 有助于提升网络服务质量、单位时间内的云服务商收益以及 VM 资源利用率.

根据本文算法构建的模型, 云服务商的收益可以表示为:

$$\sum_{p=1}^n P_{p'} \lambda_p - \sum_{k=1}^K \sum_{i=1}^N \sum_{p=1}^n C_{k,i,R} \lambda_p - \sum_{k=1}^K \sum_{i=1}^N \sum_{p=1}^n C_{k,i,I} (1 - \lambda_p) \quad (5)$$

其中: λ_p 为服务应答系数, 当云服务商接受用户服务请求时, $\lambda_p = 1$; 否则 $\lambda_p = 0$.

显然, 用户群 U 还应满足以下条件:

$$t_p \in [t_{\min}, t_{\max}] \quad (6)$$

$$\sum_{p=1}^n b_p \leq B \quad (7)$$

第 1 节中提出的 4 种动态资源信息描述指标——资源需求量、处理能力、负荷指数、成本指数可以用来对云网络进行性能评估, 从而为 VM 资源的配置提供科学依据. 由 4 种指数获取的综合指标为性能指数 (PI), PI 数值介于区间 $[0, 1]$, PI 数值越大, 表明云服务商的 VM 资源越紧张, 云性能越差, PI 的数学表达式为:

$$PI = \begin{cases} 1, & \text{如果 } RRQ_i = R^* \text{ 或 } DA_i \geq D^* \text{ 或 } LI_i \geq L^* \text{ 或 } CI_i \geq C^* \\ \max \left[\frac{RRQ_i}{R^*}, \frac{DA_i}{D^*}, \frac{LI_i}{L^*}, \frac{CI_i}{C^*} \right], & \text{否则} \end{cases} \quad (8)$$

其中: RRQ_i , DA_i , LI_i , CI_i 分别表示当前 VM 资源的服务请求量、单位时间内完成的服务请求量、负荷指

数以及成本指数； R^* ， D^* ， L^* ， C^* 分别表示当前 VM 资源所能承受的最大服务请求量、单位时间内能够完成的最大服务请求量、能够承受的最大负荷以及云服务商能承受的最大成本。需要指出的是， \max 函数中的 4 个参数均需要进行归一化处理。

式(8)表明，当 VM 资源的 4 项指标中任一项达到最大值时， PI 数值为 1，表明此时云中的 VM 资源已经非常紧张；反之，表示云中的 VM 资源仍有结余可供使用。

VM 资源配置算法根据用户群服务请求和云性能指数共同确定。算法的具体步骤如下：

输入：云性能指数 PI_i ，云服务商拥有的 VM 资源总量折算为 N ，单位 VM 资源 $S_i (1 \leq i \leq N)$ 的运行维护成本和闲置维护成本分别为 $C_{i,R}, C_{i,I}$ ，某时刻公布的 VM 资源竞拍时限为 $[t_{\min}, t_{\max}]$ ，网络总带宽资源为 B ，用户 $U_p (1 \leq p \leq n)$ 对云服务商提出的服务请求为 $(S_{i,p}, b_p, t_p, P_p)$ ；

输出：竞拍成功的用户群体集合 S_U 以及对应获得的 VM 数量 $VM_k, 1 \leq k \leq K$ 。

初始条件：筛选后的竞拍用户群集合 $S_U = \Phi$ ， $VM_k = 0, 1 \leq k \leq K$ ，公布某时刻的 VM 资源竞拍时限为 $[t_{\min}, t_{\max}]$ ，筛选后的竞拍用户群个体计数器 $q = 0$ 。

Step1: 计算云计算指数 PI_i

Step2: $N = N * (1 - PI_i)$; $B = B * (1 - PI_i)$

Step3: for ($p = 1$; $p \leq n$; $p++$); $q++$

Step4: if $t_p \leq t_{\max}$

Step5: 筛除用户 U_p

Step6: else $U'_q \leftarrow U_p$

Step7: end if else; end for

Step8: 按照用户服务请求中声明所能承担的最高价格 P 对筛选后剩余的 q 个用户进行排序

Step9: for ($x = 1$; $x \leq q$; $x++$)

Step10: if $P_x < C_{k,i,R} - C_{k,i,I}$ or $b_x > B$ or $S_{i,p} > N$

Step11: 筛除用户 U'_x

Step12: 保留用户 U'_x

Step13: else $S_u = S_u \cup U'_x, B = B - b_x, N = N - S_{i,p}$

Step14: end if else; end for

Step15: 对剩余的 x 个竞拍成功用户按照服务请求中声明所能承担的最高价格 P 进行资源分配。

在上述算法中，云服务商共进行了 3 次筛选过程。第一次筛选针对竞拍时刻展开，主要解决用户群中可能出现的下单请求时刻与最终竞拍时刻间隔时间过长的的问题。以往 VM 资源调度算法中往往没有考虑竞拍时限问题，对这一问题的忽略容易造成一些“不诚心购买资源”的用户长期占有潜在资源，使得另一些“急需购买资源”的用户无法成功申请资源的情况出现，该情形类似于淘宝中的“已拍下”与“买家已付款”情况；第二次筛选针对用户群的竞拍价展开，云服务商维护 VM 资源需要一定的成本，因而云服务商提供服务的前提在于用户群竞拍的支付价格必须要高于云服务商资源维护的成本，故需要对满足竞拍时限要求的用户个体进行支付价格的筛选；第三次筛选针对云服务商所能提供的资源总量展开，如果用户群体的带宽需求或是 VM 资源数量需求超过云服务商所能提供的程度，该用户服务请求必须被拒绝。此外，该算法通过对评估云性能得出云计算指数 PI 从而实现对云服务商资源的合理调整，进一步优化云端服务性能。

用户在进入竞拍成功的用户群体集合 S_U 后，即将面临对所申请的服务请求付费的问题，在付费时需要综合考虑多重因素，例如竞争对手支付的服务价格，以及用户群体集合 S_U 中的平均支付水平。为了综合确保用户能够尽可能争取到自己所需的资源同时兼顾云服务商的收益水平，本节提出一种用户竞拍支付算法，假设用户群体集合 S_U 的元素个数为 x ，算法的具体步骤如下：

输入: 竞拍成功的用户群体集合 S_U , S_U 中的个体为 $s_u (1 \leq u \leq x)$, 集合中的每个个体的竞拍价格为 $P_U (1 \leq U \leq x)$, 性能指数 PI_i , 云服务商拥有的 VM 资源总量折算为 N ;

输出: 竞拍成功用户的最终支付价格 $P_{u'}$;

Step1: for ($u = 1; u \leq x; u++$)

Step2: 针对用户 s_u , 令 S_{U-u} 表示 S_U 去掉 s_u 后的成员构成的集合, 并计算集合 S_{U-u} 中各用户个体的平均支付水平 P_{U-u}

Step3: 得出对用户 s_u 竞争力最大的用户 s_z 支付的服务价格 P_z

Step4: $P_{u'} = \max(P_{U-u}, P_z)$; 令 P_{U-u} 表示集合 S_{U-u} 中各用户个体的平均支付水平

Step5: end for

在以上算法中, 云服务商将对进入竞拍成功用户群体集合 S_U 中的个体进行研究, 讨论其支付问题. 本算法中涉及对除本用户个体之外群体的平均支付水平以及与本用户个体竞争最为激烈的个体的支付价格进行研究. 其中平均支付水平反映了群体中的宏观支付能力, 相当程度上描述了整个用户群体集合的支付水平; 竞争最为激烈的个体即本用户个体退出后, 最有可能获得相应 VM 资源的个体, 该个体的支付水平也应该成为讨论的重点. 因此, 为了同时确保本用户个体能够取得 VM 资源以及云服务商的收益水平, 应当使本用户个体的最终支付价格为平均支付水平以及竞争最为激烈的个体支付价格二者中的最大值.

2.2 基于迁移机制的资源调配模型

通常情况下, 用户对 VM 资源的申请需求仅停留在量的层次, 而不会特定对指定的 VM 资源进行申请. 然而, 在特殊条件下, 特定的用户可能会对特殊的云资源进行申请, 例如对网络打印机或扫描仪资源的申请. 为了保证用户服务质量和最大限度提升云服务商的运营效益, 有必要对云环境下的资源进行实时评估, 将 VM 资源划分为空闲 (Idle)、正常 (Normal) 和过载 (Overload) 3 种情况, 并及时地将空闲和过载资源加以迁移 (Transfer). 可以构建云服务商 VM 资源基本模型:

云服务商 S_{cloud} : 拥有 VM 资源种类数为 K , 第 k 类中的 VM 资源总量为 $x_k (1 \leq k \leq K)$.

VM 资源动态调配算法的具体步骤如下:

输入: 云服务商 VM 资源 $S_{k,i} (1 \leq k \leq K, 1 \leq i \leq x_k)$, 用户 $U_p (1 \leq p \leq n)$ 对云服务商提出的服务请求 A_{pk} ; 第 k 类中的 VM 资源剩余量为 $y_k (1 \leq k \leq K)$.

输出: VM 资源状态评估结果 VM_{state} .

初始条件: 每一类 VM 资源的原始数量均为 $x_k (1 \leq k \leq K)$, $y_k = x_k, \lambda = 0.2$.

Step1: for ($p = 1; p \leq n; p++$)

Step2: if $A_{pk} < y_k; y_k = y_k - A_{pk}$

Step3: 将 VM 资源分配给对应用户 U_p

Step4: else $A_{pk} \geq y_k; y_k = 0; VM_{state} = \text{Overload}$

Step5: Transfer

Step6: for ($k = 1; k \leq K; k++$)

Step7: if $y_k/x_k < \lambda; VM_{state} = \text{Overload}; \text{Transfer}$

Step8: else if $y_k/x_k > 1 - \lambda; VM_{state} = \text{Idle}; \text{Transfer}$

Step9: else $VM_{state} = \text{Normal}$

Step10: else end if else; end for.

利用上述算法可以对云环境下的 VM 资源状态进行实时评估, 将 VM 资源划分为空闲 (Idle)、正常 (Normal) 和过载 (Overload) 3 种情况. 根据熵理论, VM 资源处于正常状态时最有助于在保证用户 QoS 的前提下提升云服务商的运营效益; 相反, 当 VM 资源处于空闲时, 大量云 VM 资源处于闲置状态, 显然造成

了资源的浪费,不利于提升云服务商运营效益;类似地,当云 VM 资源处于过载状态时,大量用户请求得不到及时处理,因而也不利于增加云服务商收益.因此,在空闲和过载两种状态下,云服务商应尽可能对 VM 资源进行迁移处理,提升资源利用率,增加运营效益.

3 仿真实验与分析

为了验证本文算法的合理有效性,本节将采用 CloudSim3.0 软件在 Linux 系统平台上进行实验操作,实验平台的配置为 CPU 2.8GHz,8GB 内存,1TB 硬盘.

3.1 基于网络带宽和拍卖机制的支付算法仿真

该组仿真实验参数设置如下:云服务商提供的 VM 资源总量 N 折算为 40 000,用户总数 n 为 20 000,网络带宽资源总量上限 B 为 11 Mbit/s,单位 VM 资源的运行维护成本和闲置维护成本分别为 0.5 cent/h 和 0.15 cent/h.

此外,本节还选取了 3 种代表性算法与本文算法进行比较,其中包括:

算法 1: 基于拍卖的 VM 动态分配与供给联合策略^[21];

算法 2: 固定价格 VM 资源分配策略;

算法 3: VM 动态调度算法^[11].

需要说明的是,本文算法引入了竞拍时限机制,因此在仿真实验过程中需要由软件设定 VM 资源竞拍时限 $[t_{\min}, t_{\max}]$,为了更客观地对算法 2 进行描述,本节在进行仿真实验时将设定两个固定数值,分别为:0.45 cent/h,0.30 cent/h.

图 2 给出了本文算法在某一时刻的云服务商收益与竞拍时限间隔之间的关系图.图中不难看出,当用户数量较少时,本文算法在竞拍时限间隔较长的情况下云服务商收益较高,原因在于能够为少量的用户提供更充裕的时间参与 VM 资源需求申请;然而,当用户数量增加时,由于可能存在部分用户未能及时支付费用而长期占据 VM 资源申请序列,因此竞拍时限间隔过长反而会在一定程度上影响云服务商收益.从仿真实验结果看,本实验环境下竞拍时限间隔 $t_{\max} - t_{\min}$ 设定为 20 min 较为合适,后文中如无特别说明,默认竞拍时限间隔为 20 min.

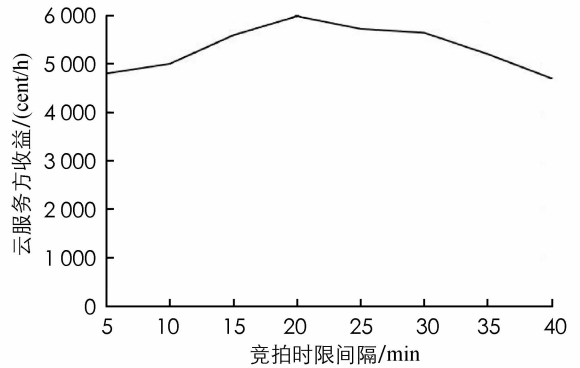


图 2 某时刻云服务商收益与竞拍时限间隔关系图

图 3 给出了包括本文算法在内的 4 种算法的云服务商收益与用户数量之间的关系图.算法 1 云服务商收益与算法 2 中固定价格为 0.30 cent/h 的收益走向趋势基本接近,均体现出稳定、平缓的趋势,表现出很好的鲁棒性和网络适应性;而算法中固定价格为 0.45 cent/h 的收益走向趋势却与 0.30 cent/h 的对应图有较大差异,尤其是在用户数量小于 14 000 时云服务商收益为负数,原因在于主观上愿意承担 0.45 cent/h 固定价格的用户量过少,从而导致 VM 资源闲置维护成本总量过大.算法 3 由于引入了临界支付功能导致云服务商收益水平变化幅度较大.本文算法由于综合考虑了用户群平均支付水平以及竞争个体支付水平,因此,云服务商收益波动幅度较算法 3 缓和.此外,综合考虑多重因素的用户竞拍支付算法提升了云服务商的收益水平.

图 4 给出了云服务商收益与带宽资源的关系图.

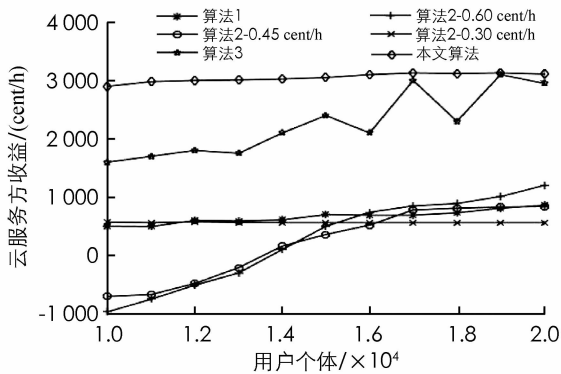


图 3 云服务商收益与用户数量关系图

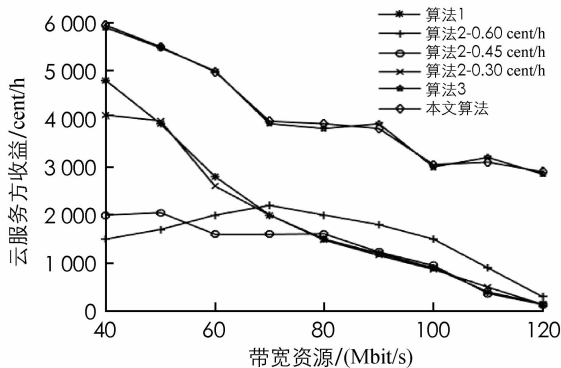


图 4 云服务商收益与带宽资源关系图

与图 3 中的情况类似, 算法 1 和算法 2 在图 4 中仍表现出良好的鲁棒性, 而算法 3 的曲线幅度波动较大. 值得注意的是, 4 种算法的云服务商收益均呈现出下降趋势, 原因在于用户带宽要求的提升必然会导致可供分配 VM 资源数量的下降, 从而影响收益. 本文算法和算法 3 都将带宽资源考虑在内, 因而均倾向于优先应答带宽资源要求较低、VM 资源要求数量较少、支付价格较高的用户服务请求.

3.2 基于迁移机制的资源调配模型的算法仿真

该组仿真实验参数设置如下: 任务长度即数据传输量取值范围为[800, 1500], 节点处理速度取值范围为[15, 180], 负载初始值取值范围为[0.01, 0.1], 故障率取值范围为[0.001, 0.01], 云环境节点数量为 1000.

此外, 本节还选取了两种代表性算法与本文算法进行比较, 其中包括:

算法 1 (highly available job execution service, HA-JES)^[22], 该技术可将任务集中到一个或多个资源之上, 以提升资源利用率;

算法 2 (availability check technique, ACT)^[23], 仅当所有资源可用时方可调度任务.

本节将针对包括本文算法在内的 3 种算法进行云资源利用率指标的检测, 该指标值为正在运行中的节点数目占总节点数量的比值, 比值越大代表云环境下的 VM 资源利用效率越高. 图 5 给出了在不同节点数条件下的云资源利用率的仿真结果.

从图 5 不难看出: 一方面, 随着节点数量的增加, 3 种算法的云资源利用率都出现了下降, 但相对 HA-JES 和 ACT 两种算法, 本文算法的资源利用率明显高于前两者, 体现出了算法良好的鲁棒性; 另一方面, 随着节点数量的增加, 本文算法始终保持较高的云资源利用率, 意味着云服务商端口中较大比例的资源均处于“服务中”状态, 而“闲置”资源量较小, 因此可以在保证用户服务质量的同时创造较高的收益.

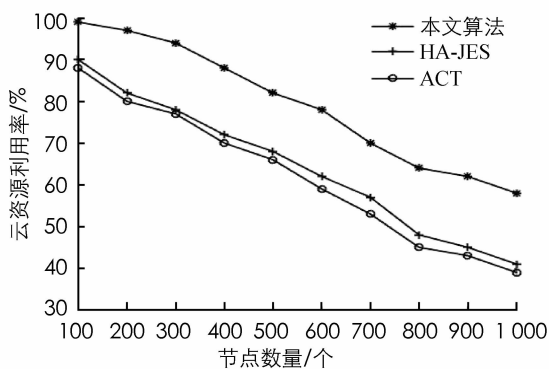


图 5 云资源利用率与节点数量关系图

综上所述, 相比其他几种算法, 本文算法能够在满足用户资源需求、降低云服务商运营成本的基础上实现云环境资源的自适应合理配置.

4 结 语

本文在传统云计算 VM 资源调度算法的基础上, 综合考虑了网络带宽以及竞拍机制的特点, 一方面筛选出了符合用户服务质量和云服务商最低成本的资源调度方案; 另一方面, 也实现了对云服务商 VM 资源

的动态评估和整体把握, 实现了 VM 资源的合理应用。仿真实验结合两方面内容均进行实验, 结果表明本文算法在保证用户服务质量和提升云服务商运营效益方面均有显著的优势。

参考文献:

- [1] 王 倩, 石振国, 孙万捷, 等. 基于 PEPA 的云计算资源分配算法性能评价 [J]. 计算机应用研究, 2015, 32(4): 1179—1183.
- [2] TIAN F G, CHEN K K. Towards Optimal Resource Provisioning for Running Mapreduce Programs in Public Clouds [C]//Proceedings of the IEEE International Conference on Cloud Computing. New York: IEEE Computer Society Press, 2011: 155—162.
- [3] HE S J, GUO L, GUO Y K, et al. Elastic Application Container: A Lightweight Approach for Cloud Resource Provision [C]//Proceedings of the 2012 IEEE 26th Conference on Advanced Information Networking and Applications (AINA). New York: IEEE Computer Society Press, 2012: 15—22.
- [4] QUIROZ A, KIM H, PARASHAR M, et al. Towards Autonomic Workload Provisioning for Enterprise Grids and Clouds [C]//Proceedings of the 10th IEEE/ACM International Conference on Grid Computing. New York: IEEE Computer Society Press, 2009: 50—57.
- [5] VERMA A, CHERKASOVA L, CAMPBELL R H. Resource Provisioning Framework for MapReduce Jobs with Performance Goals [J]. Lecture Notes in Computer Science, 2011, 70(9): 165—186.
- [6] GONG Z H, GU X H, WILKES J. PRESS: Predictive Elastic Resource Scaling for Cloud Systems [C]//Proceedings of the 2010 International Conference on Network and Service Management (CNSM). New York: IEEE Computer Society Press, 2010: 9—16.
- [7] 胡周君. 计算网格中面向 QoS 的资源可用性评估模型研究 [D]. 长沙: 中南大学, 2010.
- [8] CHOHAN N, CASTILL C, SPREITZER M, et al. See Spot Run: Using Spot Instances for Mapreduce Workflows [C]//Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing. New York: IEEE Computer Society Press, 2010: 7—7.
- [9] ZAMAN S, GROSU D. Combinatorial Auction-based Dynamic Vm Provisioning and Allocation in Clouds [C]//Proceedings of the 3rd IEEE International Conference on Cloud Computing Technology and Science (CloudCom). New York: IEEE Computer Society Press, 2011: 107—144.
- [10] ZAMAN S, GROSU D. Combinatorial Auction-based Allocation of Virtual Machine Instances in Clouds [J]. Journal of Parallel and Distributed Computing, 2013, 73(4): 495—508.
- [11] 柳 兴, 袁超伟, 杨 震, 等. 面向移动云计算的 VM 动态调度算法 [J]. 系统工程与电子技术, 2015, 37(9): 2176—2181.
- [12] 左利云, 曹志波, 董守斌. 云计算虚拟资源的熵优化和动态加权评估模型 [J]. 软件学报, 2013, 24(8): 1937—1946.
- [13] GARRY S, MARK B. A Flexible Monitoring and Notification System for Distributed Resources [C]//Proceedings of the 7th International Symposium on Parallel and Distributed Computing. New York: IEEE Computer Society Press, 2008: 31—38.
- [14] LIU Y, GAO S. WSRF-based Distributed Visualization [C]//Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid. New York: IEEE Computer Society Press, 2009: 615—619.
- [15] FIGUEIREDO R. Adaptive Predictor Integration for System Performance Prediction [C]//Proceedings of IEEE International Parallel and Distributed Processing Symposium. New York: IEEE Computer Society Press, 2007: 1—10.
- [16] DIAZ I, FERNANDEZ G, MARTIN M. Integrating the Common Information Model with MDS4 [C]//Proceedings of the 9th IEEE/ACM International Conference on Grid Computing. New York: IEEE Computer Society Press, 2008: 298—303.
- [17] KATSAROS G, KOUSIOURIS G, GOGOUVITIS S V, et al. A Self-Adaptive Hierarchical Monitoring Mechanism for Clouds [J]. Journal of Systems and Software, 2012, 85(5): 1029—1041.
- [18] GONG ZH, GU XH, WILKES J. PRESS: Predictive Elastic Resource Scaling for Cloud Systems [C]//Proceedings of

the 2010 International Conference on Network and Service Management (CNSM). New York: IEEE Computer Society Press, 2010: 9–16.

- [19] 郑 卉, 郭 平, 李 琪, 等. 基于带宽约束的云计算负载调度算法 [J]. 西南师范大学学报(自然科学版), 2014, 39(7): 121–128.
- [20] 王 波. 基于遗传算法的集群虚拟机资源调度研究 [J]. 西南师范大学学报(自然科学版), 2016, 41(3): 107–111.
- [21] ZAMAN S, GROSU D. Combinatorial Auction-based Allocation of Virtual Machine Instances in Clouds [J]. Journal of Parallel and Distributed Computing, 2013, 73(4): 495–508.
- [22] BUYYA R, MURSHED M. Gridsim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing [J]. Concurrency and Computation: Practice and Experience, 2002, 14(12): 1175–1220.
- [23] AZOUGAGH D, YU JL, KIM JS, et al. Resource Co-Allocation: A Complementary Technique that Enhances Performance in GridComputing Environment [C]//Proceedings of the 11th International Conference on Parallel and Distributed System. New York: IEEE Computer Society Press, 2005: 36–42.

Resource Scheduling Algorithm in Cloud Computing Based on Adaptive Auction Mechanism

WANG Rui-na

Information Institute of Media, Sanmenxia Polytechnic, Sanmenxia Henan 472000, China

Abstract: For sake of overcoming the unreasonable issues of resource scheduling in cloud computing, a novel resource scheduling algorithm in cloud computing based on adaptive auction mechanism has been proposed in this paper. Firstly, by combining the characteristics of network width and auction mechanism, the new customers within the auction deadline are sorted by the proposed algorithm, and the resource scheduling scheme meeting the quality of service (QoS) and the lowest costing of cloud service providers can be screened out. Secondly, the reasonable usage of resource can be achieved by transferring the requests of hot resources. Simulation experiments demonstrate that the adaptively reasonable resources configuration in cloud computing can be realized by the proposed novel algorithm, which can satisfy the needs of the resource demands from the customers and decrease the operating costs of the cloud service providers.

Key words: cloud computing; resource; auction; transferring

责任编辑 张 枸
崔玉洁