

云计算中基于拍卖的虚拟机动态供应和分配算法

刘钟涛 刘明利

(河南财政政法大学现代教育技术中心 郑州 450046)

摘要 当前云计算供应商通过定价算法或类似拍卖的算法来分配他们的虚拟机(VM)实例。然而,这些算法大多要求虚拟机静态供应,无法准确预测用户需求,导致资源未得到充分利用。为此,提出了一种基于组合拍卖的虚拟机动态供应和分配算法,在做出虚拟机供应决策时考虑用户对虚拟机的需求。该算法将可用的计算资源看成是“流体”资源,且这些资源根据用户请求可分为不同数量、不同类型的虚拟机实例。然后可根据用户的估价决定分配策略,直到所有资源分配完毕。基于并行工作负载存档(Parallel Workload Archive)的真实工作负载数据进行了仿真实验,仿真结果表明所提方法可保证为云供应商带来更高收入,提高资源利用率。

关键词 云计算,虚拟机实例,拍卖,分配,云供应商,资源利用率

中图分类号 TP391 文献标识码 A

Virtual Machine Dynamic Supply and Allocation Algorithm Based on Auction in Cloud Computing

LIU Zhong-tao LIU Ming-li

(The Modern Educational Technology Center, Henan University of Economics and Law, Zhengzhou 450046, China)

Abstract Current cloud computing providers allocate their virtual machine (VM) instances via fixed price-based or auction-like mechanisms. However, most of these algorithms require static supply virtual machine, and they are unable to accurately predict the user demand, lead to underutilization of resources. To this end, an auction-based algorithm for dynamic VM provisioning and allocation was proposed that takes into account the user demand for VMs when making VM provisioning decisions. The algorithm treats the set of available computing resource as ‘liquid’ resources that can be configured into different numbers and types of VM instances depending on the requests of the users, and the proposed algorithm determines the allocation strategy based on the users’ valuations until all resources are allocated. Our mechanism is evaluated by performing simulation experiments using traces of real workload from parallel workload archive, the results show that the proposed method can guarantee to bring the higher income for cloud providers, and improve the resource utilization rate.

Keywords Cloud computing, Virtual machine instances, Auction, Allocation, Cloud providers, Resource utilization rate

1 引言

云计算系统所提供的未来计算基础设施使得用户可以为他们的计算需求分配更多的远程资源,节约了自己建立系统所需的前期成本。云供应商将他们的资源以多种类型虚拟机(VM)实例的方式供应,然后使用定价策略来分配和出售 VM 实例^[1],并且使用拍卖算法来出售定价之后仍然闲置的资源。云供应商希望分配算法能够支持动态供应,以便他们根据市场需求就不同类型的 VM 实例数量做出决策。

多篇文献对虚拟机的动态供应和分配问题进行了研究。文献[2]设计了一种在线虚拟机动态供应和分配算法,在该算法中每当有足够多的资源和合适的申请时,就会分配 VM 实例,然而该算法在运行时需要收集一段时间内所有用户请求的所有信息;文献[3]为了改善云采购平台中的顾客满足情况,提出了载有虚拟供应商资源的虚拟机分配流程,并且建立了模型,然后分别采用最佳递减匹配(BFD)方法和仪跟踪多

群粒子群优化(FTMPSO)算法对其进行求解,然而该方法还存在着较多不足,例如对需求到达的模拟过于粗糙,对虚拟机迁移的成本没有过多的考虑;文献[4]分析和研究了 Xen 虚拟机管理器的 Credit 调度算法的不足,提出了改进的调度算法对虚拟机进行 Credit 比例预分配,采用动态调度时间片机制,以 non-work-conserving 方式实现软实时任务周期调度,然而 Credit 值的更新会受到更新时间片长度、虚拟机数量以及虚拟机权重等因素的影响,响应性能受到限制,系统的公平性无法保证;文献[5]提出了基于组合拍卖的近似策略(CA-GREEDY),并证明了基于组合拍卖的机制是虚拟机拍卖问题的最优求解策略,虽然该策略可以提高虚拟机实例的分配效率及云供应商的收入,但是该策略要求虚拟机实例已经供应完毕且不会再变,资源未得到充分利用。

为此,本文正式描述了虚拟机动态供应和分配问题,提出一种基于组合拍卖的求解算法,该算法可保证为云供应商带来更高收入,提高资源利用率。文中还分析了运行本文算法

本文受河南省教育厅高等学校重点科研项目(15A510039,16A510024)资助。

刘钟涛(1980—),男,硕士,工程师,主要研究方向为云计算、数据挖掘;刘明利(1981—),男,硕士,高级工程师,主要研究方向为大数据、云计算。

时的成本和效益,并给出了部署指导原则。最后基于并行工作负载存档(Parallel Workload Archive)^[6]的真实工作负载数据进行仿真实验,评估了本文方法的有效性。

2 虚拟机动态供应和分配问题

本文将虚拟机动态供应和分配问题(DVMPA)定义如下。设云供应商通过 m 种不同类型的虚拟机实例 VM_1, \dots, VM_m 向用户提供计算服务。类型为 $VM_i (i=1, \dots, m)$ 的虚拟机实例的计算能力为 w_i , 其中 $w_1=1, w_1 < w_2 < \dots < w_m$ 。本文用 $w=(w_1, w_2, \dots, w_m)$ 表示 m 种虚拟机的计算能力向量。下文中使用该向量表示“权重向量”。假设一个云供应商提供 3 种类型的虚拟机实例: VM_1 , 包括 1 个 2GHz 处理器, 4GB 内存, 500GB 存储空间; VM_2 , 包括 1 个 2GHz 处理器, 8GB 内存, 1TB 存储空间; VM_3 , 包括 2 个 GHz 处理器, 16GB 内存, 2TB 存储空间。描述这 3 种虚拟机实例的权重向量为 $w=(1, 2, 4)$ 。本文假设云供应商的所有资源可创建最多 M 个性能最低的 VM_1 实例, 云供应商根据 VM_1, \dots, VM_m 明确的类型可用多种方式供应虚拟机实例。设 k_i 表示云供应商提供的 VM_i 类型实例的数量, 只要 $\sum_{i=1}^m w_i k_i \leq M$, 那么供应商就可供应 (k_1, k_2, \dots, k_m) 确定的各种实例组合。

假设 n 个用户 u_1, \dots, u_n 向云供应商请求批量虚拟机实例计算资源。用户通过向云供应商提交 $B_j=(r_1^j, \dots, r_m^j, v_j)$ 申请来请求虚拟机实例, 其中 r_i^j 表示 VM_i 类型实例的数量, v_j 表示用户 u_j 为了在单位时间内使用请求的批量虚拟机所愿意支付的费用, 比如用户向提供 3 种类型虚拟机的云供应商提交的申请可为 $B_j=(2, 1, 4, 10)$, 该申请表示用户为了在单位时间内使用 2 个 VM_1 实例、1 个 VM_2 实例和 4 个 VM_3 实例, 同时愿意支付 10 个货币单位。本文让供应商周期性地运行一种拍卖机制(比如每小时运行一次)来供应和分配虚拟机实例, 实现收益最大化。为了定义云供应商的收益, 需要引入其它标记法。文中用 p_j 表示用户 u_j 为其申请的批量虚拟机实例所支付的费用, 请注意, 随着供应商使用的定价和分配不同, p_j 和 v_j 也会不同, 且往往 $p_j \leq v_j$ 。

假设连续两次拍卖的时间间隔为一个时间单位。设 c_R 和 c_I 表示一个 VM_1 实例在一个单位时间内处于运行状态和空闲状态时的成本。很显然, 有 $c_R > c_I$ 。云供应商运行所有可用资源的成本(即所有 M 个 VM_1 实例)为 $M \cdot c_R$, 而使所有可用资源处于空闲状态的成本为 $M \cdot c_I$ 。本文用 $x=(x_1, x_2, \dots, x_n)$ 表示分配向量, 其中如果用户 u_j 请求的批量实例被分配给该用户, 则 $x_i=1$, 否则 $x_i=0$ 。已知一个具体的分配向量和支付费用, 云供应商的收益为:

$$P = \sum_{j=1}^n x_j p_j - c_R \sum_{j=1}^n x_j s_j - c_I (M - \sum_{j=1}^n x_j s_j) \quad (1)$$

其中, $s_j = \sum_{i=1}^m w_i r_i^j$, 即用户 u_j 请求的“单位”计算资源的数量。“单位”计算资源等于 1 个 VM_1 实例(即提供的性能最小的实例)。式(1)中的第 1 项表示营收; 第 2 项表示分配给用户的虚拟机实例的运行成本; 第 3 项表示剩余资源处于空闲状态的成本。综上所述, 云计算下虚拟机动态供应和分配问题可定义为:

$$\begin{aligned} & \max P \\ & \text{s. t. } \sum_{j=1}^n s_j \leq M \\ & \quad x_j \in \{0, 1\} \\ & \quad 0 \leq p_j \leq v_j \end{aligned} \quad (2)$$

该问题的解包括请求批量实例 $(r_1^j, \dots, r_m^j) (j=1, \dots, n)$ 的每个用户 u_j 的分配策略 x_j 及其价格 p_j 。分配策略需要确定需要被供应的每种类型虚拟机的数量, 具体如下文所示。对每种实例类型 VM_i 计算 $k_i = \sum_{j=1}^n x_j r_i^j$, 然后供应 k_i 个类型为 VM_i 的虚拟机实例。

文献[7]已经证明, 相比于当前的固定价格机制, 组合拍卖机制可提高虚拟机实例的分配效率和收益。然而该文献研究的组件拍卖机制要求虚拟机已经提前分配完毕。本文认为, 如果在拍卖时对虚拟机实例实行仔细的动态选择, 以反映当时的市场需求, 则可提升系统的总体性能。下节提出一种可解决 DVMPA 问题的组合拍卖算法, 通过确定云供应商需要供应的虚拟机的分配策略、价格和最优配置来提高收益。

3 基于组合拍卖的虚拟机实例动态供应和分配机制

本节提出求解 DVMPA 问题的组合拍卖算法 CAM-DVMPA。该算法可以确定获胜用户需要支付的价格以及为了满足获胜用户的要求需要供应的虚拟机实例集合。该算法可同时保证被分配的资源数量最大, 且虚拟机实例分配时价格不低于底价。

CAM-DVMPA 采用底价策略以保证用户的支付费用不低于云供应商确定的数额。因此, 云供应商需要设置底价 v_{res} , 该底价取决于虚拟机的运行成本。本文取底价为 c_R 和 c_I 间的保本点, 即 $c_R - c_I$, 这是因为如果有一个单位的资源未被分配, 则导致损失 c_I 。同时, 如果该资源分配时的价格为 $c_R - c_I$, 则损失为 $c_R - (c_R - c_I) = c_I$ 。换句话说, 用户为了获得性能最低虚拟机在一个单位时间内的使用权所必须支付的费用, 等于该资源运行成本和空闲成本之差。支持底价策略 v_{res} 的拍卖, 可看成是引入虚拟投标人 u_0 的不支持底价策略拍卖, 其中虚假投标人以底价作为其估价, 即 $v_0 = v_{res}$ 。虚拟用户 u_0 出价 $B_0=(1, 0, \dots, 0, v_{res})$, 即对所有 $i=2, \dots, m$ 有 $i=2, \dots, m$ 且 $v_0 = v_{res}$ 。CAM-DVMPA 根据出价的密度来确定分配策略。用户 u_j 的出价密度为 $d_j = v_j / s_j$, 其中 $s_j = \sum_{i=1}^m w_i r_i^j$ ($j=0, \dots, n$)。出价密度衡量用户为使用一个单位的分配资源所支付的费用。在本文中, 分配到的单位资源对应于一个 VM_1 实例, 为了保证用户至少支付底价, 本文机制将排除 $d_j < d_0$ 的所有用户。

算法 1 CAM-DVMPA 算法

输入: $M; m; w_j; j=1, \dots, n; c_R, c_I$;

输出: $W; p_j; j=1, \dots, n; k_i; i=1, \dots, m$;

1. {阶段 1: 收集出价}
2. for $j=1, \dots, n$ do
3. 收集用户 u_j 的出价 $B_j=(r_1^j, \dots, r_m^j, v_j)$
4. end for
5. {阶段 2: 确定获胜用户和供应策略}
6. $W \leftarrow \emptyset$ {获胜用户集合}
7. $v_{res} \leftarrow c_R - c_I$

8. 增添出价为 $B_0 = (1, 0, 0, \dots, 0, v_{res})$ 的虚拟用户 u_0
9. for $j=0, \dots, n$ do
10. $s_j \leftarrow \sum_{i=1}^m r_i^j w_i$
11. $d_j \leftarrow v_j / s_j$ {‘出价密度’}
12. end for
13. 对用户 u_1, \dots, u_n 重新排序, 于是有 $d_1 \geq d_2 \geq \dots \geq d_n$
14. 设 l 表示索引, 于是当 $j \leq l$ 时 $d_j \geq d_0$, 否则 $d_j < d_0$
15. 排除用户 u_{l+1}, \dots, u_n
16. 将用户 u_0 重命名为 u_{l+1}
17. 设置 $n \leftarrow l+1$
18. $n \leftarrow l+1$
19. for $j=1, \dots, n-1$ do {排除虚拟用户}
20. if $s_j \leq R$ then
21. $W \leftarrow W \cup u_j$
22. $R \leftarrow R - s_j$
23. end if
24. end for
25. for $i=1, \dots, m$ do {确定虚拟机配置}
26. $k_i \leftarrow \sum_{j: u_j \in W} r_i^j$
27. end for
28. {阶段 3: 支付}
29. for 所有的 $u_j \in W$ do
30. $W_j' \leftarrow \{u_l : u_l \in W \rightarrow u_l \in W\}$
31. $l \leftarrow W_j'$ 中的最小索引
32. $p_j \leftarrow d_l s_j$
33. end for
34. for 所有 $u_j \notin W$ do
35. $p_j \leftarrow 0$
36. end for
37. 返回 (W, p, k)

CAM-DVMPA 的具体过程见算法 1。该算法需要部分系统信息, 以便云供应商在供应资源时将计算资源总量 M 表示为 VM_1 虚拟机总量。该机制同时需要可用的虚拟机类型数 m 及其权重向量 w , 还需要知道 VM_1 虚拟机实例的运行成本 c_R 及空闲成本 c_I 。

CAM-DVMPA 分为 3 个阶段。在第 1 阶段, 它收集用户的出价 B_j (第 1—4 行)。在第 2 阶段, 该机制确定云供应商需要供应的虚拟机配置及获胜出价方, 具体过程如下: 它增加一个虚拟用户 u_0 , 出价内容包括 1 个 VM_1 实例, 出价为 $v_{res} = c_R - c_I$ (第 8 行)。该虚拟用户只用于模拟支持底价的拍卖模型, 不会被分配任何资源。然后, 它将计算所有用户的批量资源规模 s_j 以及出价密度 d_j (第 9—12 行)。此后, 除了虚拟用户外的所有用户按照出价密度降序排列, $d_j < d_0$ 的所有用户被删除 (第 13—15 行)。然后, 它将虚拟用户 u_0 移至剩余用户的列表末尾, 因为虚拟用户在当前的用户集合中出价密度最低, 将包括虚拟用户在内仍被考虑的用户总量重新设置为 n (第 16 行、17 行)。此后, 算法采用贪婪策略确定获胜用户。只要有资源可用, 那么便按照出价密度降序次序向用户分配他们所请求的批量资源, 然而在分配时不考虑虚拟用户。确定了获胜用户后, 该算法将聚集获胜用户请求的批量资源, 确定需要分配给用户们的虚拟机配置 (第 25—27 行)。在第 3 阶段, 算法确定所有用户的支付价格, 对每个获胜用户 u_j , 算法寻找失败用户 u_l (如果 u_j 不参与, 则 u_l 将获胜)。然后, 用

户 u_j 的支付费用计算为其批量资源规模 s_j 与 u_l 的出价密度之积。所有失败用户支付费用为 0, 这种支付类型通常称为临界支付。

4 理论分析

本文认为, 如果用户通过为批量虚拟机表达自己的真实估价来实现其效用最大化, 则称该机制具有真实性。其中, 用户 u_j 的效用表示为用户 u_j 对批量虚拟机的估价 v_j 与为了使用该批虚拟机而向分配机制提交的支付费用 p_j 之差。因为参与真实的分配机制的用户必须采用复杂的出价策略才能使其效用最大, 所以上述特点非常重要。他们只需为批量虚拟机提交自己的真实估价即可。文献[8]指出, 如果分配功能具有单调性, 且支付过程属于临界支付, 则该分配机制具有真实性。该描述有助于证明定理 1。

定理 1 CAM-DVMPA 是真实的

证明: 首先证明 CAM-DVMPA 分配具有单调性。用户通过增加其出价的密度即可增加申请成功的概率, 提交的估价越大, 则密度越大; 批量虚拟机规模越大, 则密度越小。于是, 用户若想在 CAM-DVMPA 进行分配决策时用到的排序表上位置靠前, 只能提高出价或降低批量虚拟机申请规模。因此, CAM-DVMPA 分配具有单调性。其次, 根据 CAM-DVMPA 的支付方式, 用户需要为其申请的批量虚拟机支付一定费用, 以便单位虚拟机实例的平均价格等于其临界支付费用。用户只是支付了赢取申请所需要的最小费用, 单调分配和临界支付特点保证 CAM-DVMPA 具有真实性。底价不会影响分配机制的真实性, 因为底价基本只是虚拟用户的出价, 而虚拟用户受云供应商控制, 所以真实出价仍然是用户的主要策略。证毕。

下面分析 CAM-DVMPA 的复杂性。算法 1 的主要计算负载来自第 19—24 行和第 29—33 行循环。第 1 个循环的最大复杂性为 $O(M)$, 即: 所有获胜用户所申请的批量实例均只包含 1 个单位的 VM_1 实例。第 29—33 行循环的总运行时间为 $O(n)$ 。这是因为它对获胜用户进行循环操作, 对失败用户进行搜索操作。因为已经对用户排序, 所以获胜用户 u_{j+1} 的临界支付搜索实际上是从 u_j 的“临界支付用户 u_l ”处开始 (不失一般性, 假设此时 u_j 和 u_{j+1} 为获胜方)。因此, 该循环的总体最大复杂性为 $O(n)$, 其中第 13 行的排序复杂度为 $O(n \log n)$ 。于是, CAM-DVMPA 机制的复杂性为 $O(M + n \log n)$ 。

5 仿真实验

本文利用真实的工作负载数据进行全面的仿真实验, 来评估 CAM-DVMPA 机制的性能。比较 CAM-DVMPA 与基于虚拟机静态分配的组合拍卖策略 CA-GREEDY^[5] 的性能。文献[5]的研究已经证明, CA-GREEDY 与当前云供应商采用的固定价格虚拟机分配机制^[9-12] 的性能相比, CA-GREEDY 的性能远优于固定价格机制, 因此在本文实验中采用该机制作为比较对象。利用并行工作负载存档^[9] 中的 11 种工作负载日志共进行 264 次实验, 其中每个工作负载有 24 种不同的参数组合。

5.1 实验配置

实验内容主要是从已知的工作负载中生成作业提交, 然

后同时运行 CA-GREEDY 和 CAM-DVMPA 来分配作业并提供虚拟机。在配置实验时还要处理工作负载选择、申请生成和建立拍卖等问题。

(1)工作负载选择:利用文献[6]中的并行工作负载存档负载。该存档包括许多网格和超级计算站点收集而来的大量工作负载。表 1 给出了工作负载的部分统计数据。

表 1 负载日志的统计数据

日志文件	时间 (小时)	每小时 作业数	平均 运行时间	每个作业的 处理器数量	处理器 数量
ANL-Intrepid-2009	5759	11.97	2.09	5063	163840
DAS2-fs0-2003	8744	25.81	1.09	10.27	144
DAS2-fs1-2003	8633	4.67	1.23	8.38	64
DAS2-fs2-2003	8760	7.58	1.29	9.45	64
DAS2-fs3-2003	8712	7.66	1.17	4.96	64
DAS2-fs4-2003	7963	4.24	1.67	3.66	64
LLNL-Atlas-2006	4308	9.92	2.52	400.7	9216
LLNL-Thunder-2007	3605	33.58	1.52	42.54	4008
LLNL-uBGL-2006	5339	21.09	1.25	575.8	2048
LPC-EGEE-2004	5728	41.01	1.80	1	140
SDSC-DS-2004	9387	10.24	2.88	62.41	1664

(2)作业和出价生成:为日志文件中的每条记录生成一个作业,用户需要运行该作业并为该作业生成一个出价。然后,需要为作业生成两个重要参数:请求的批量虚拟机和相应出价。为了生成作业的批量虚拟机实例,根据如下方法确定其通信与计算比:

$$\text{通信比} = 1 - \frac{\text{平均 CPU 时间}}{\text{总运行时间}}$$

通信比可以衡量有多少运行时间花费在给定作业的进程之间的通信上。以该值为基础,可确定作业类别,其中共有 m 种类别, m 表示可用的虚拟机类别数量。作业的类别指定了该作业的“首选”虚拟机类型。工作原理如下:用因子 μ 描述被请求的总体虚拟机中有多少属于“首选”类型的虚拟机实例。例如,请求 q_j 个处理器的类别为 i 的作业将会生成一个批量的虚拟机,该批量由一定数量的 VM_i 实例构成,以便分配 μq_j 个处理器。通过随机选择其他虚拟机类型来请求剩余处理器。生成批量虚拟机后,将生成相应的出价。为此,首先定义作业的加速因子为:

$$\text{加速因子} = \text{CPU 数量} \times \frac{\text{每个 CPU 的平均时间}}{\text{总运行时间}}$$

该加速因子与“估价因子”相乘,即可生成出价,该估价因子与用户类型有关。利用用户的 ID 号与 5 模的值来将用户划分为 5 种类别。为作业设置的最后一个参数是其截止时间,此时工作负载日志内无信息提供。假设截止时间为完成作业所需时间的 4 倍至 8 倍,因此将一个作业的截止时间设置为 4-8 间的随机数与所需时间之积。

对可以运行的作业并行且独立地运行 CA-GREEDY 和 CAM-DVMPA 机制。用户(或作业)参与拍卖,直到其作业完成或确定其作业在截止时间前无法完成。如果其作业完成,则认为用户“被服务”,否则便“未被服务”。不失一般性,假设每个用户只提交一项作业,于是在下文中“用户”与“作业”可互用。

(3)拍卖设置:假设云供应商提供 4 种不同类型的虚拟机: VM_1, VM_2, VM_3, VM_4 。这些虚拟机类型由权重向量 $w = (1, 2, 4, 8)$ 描述。对每个负载文件,提取其用户总数 N 及可

用的处理器总数 M 。随着拍卖的进行,动态确定参与一次拍卖的用户数量。

本文在生成一个用户提交的作业所需的批量虚拟机时,只配置了几个参数。向量 (C_1, C_2, C_3) 确定了用于描述作业情况的通信比。对 $(C_1, C_2, C_3) = (0.05, 0.15, 0.25)$ 使用如下,通信比低于 0.05 的作业属于类型 1,所需要的大部分虚拟机实例 μq_j 将请求 VM_1 ,其中 q_j 表示 u_j 请求的处理器数量。假设 μ 取值 0.5 和 0.75,利用其余类型的虚拟机实例来随机确定其余批量虚拟机。利用日志文件的用户 ID 字段来确定用户的估价范围。有 5 种类型的用户提交作业,用户的类型 t ,计算方法为用户 ID 与 5 模的值。日志的用户 ID 为实数,因此这种分类方法实际上可生成用户的分布。用户的每种类型 t 关联了一个“估价因子” f_t 。确定用户属于类型 t 后,利用 5.1 节讨论过的加速因子及向量 f 的“估价向量” f_t ,即可确定用户对批量虚拟机的估价。向量 f 有 5 个元素(等于用户类型数),每个元素表示该类型的用户对“每个单位的加速因子”的平均估价。具体来说,假设有用户 u_j ,其作业的加速因子为 S_j ,则如果 u_j 属于类型 t ,则该用户为了能够在一个小小时内使用其申请的批量虚拟机,平均来说愿意支付 $f_t S_j$ 。在 0 到 $2f_t$ 范围内生成一个随机数,然后与 S_j 相乘以生成均值为 $f_t S_j$ 的估价。本文使用两组 f 向量,如表 2 所列。

表 2 仿真参数

名称	描述	数值
N	总用户	来自于日志文件
M	总 CPU	来自于日志文件
T	仿真时间	来自于日志文件
(c_1, c_R)	一个单位的虚拟机的空闲和运行成本	$(0.05, 0.1), (0.1, 0.25), (0.15, 0.5)$
μ	“首选”虚拟机类型的 CPU 因子	50%, 75%
h	处理器在虚拟机类型中的静态分配	$(0.25, 0.25, 0.25, 0.25), (0.07, 0.13, 0.27, 0.53)$
f	用户类型的估价因子	$(0.5, 1, 1.5, 2, 2.5), (1, 1.5, 2, 3, 4)$
C_1, C_2, C_3	通信比的边界	$(0.05, 0.15, 0.25)$

CAM-DVMPA 机制本身就可确定云供应商需要供应的虚拟机的配置,而 CA-GREEDY 假设虚拟机静态供应,因此需要提前供应虚拟机配置。为了生成 CA-GREEDY 需要的虚拟机静态供应,使用的向量 h 如下。在仿真中假设有两个 h 实例:1) $h = (0.07, 0.13, 0.27, 0.53)$,可保证已知权重向量 w 后,每种类型的虚拟机实例数量相等或基本相等;2) $h = (0.25, 0.25, 0.25, 0.25)$,可保证所有的处理器均衡分布给不同类型的虚拟机。在表 2 中给出了所有的仿真参数。通过对数值进行全面组合,对每个日志文件实验 24 次,共计 264 次实验。

5.2 结果分析

下面分析了不同负载条件下两种算法的性能。因为工作负载在多个维度上均是异质负载,所以首先定义一个指标来描述工作负载,以便确定它们的次序,然后对分配机制的性能指标正规化,比较它们相对于工作负载特征的性能。观察表 1 中列出的负载特点可以确定,对工作负载进行比较的最优指标是正规化负载,定义如下:

$$\frac{\text{每小时的作业数量} \times \text{平均运行时间} \times \text{每个作业的平均处理器数量}}{\text{总处理器数量}}$$

每小时的作业数量乘以每个作业的平均处理器数量可以

获得每小时期间作业需要的处理器数量。然后再与平均运行时间相乘即可获得一个小时期间所有作业请求的处理器平均数量估计值。正规化负载可提供负载集合的排序。

对每组仿真实验,计算生成的总营收、总成本及每次拍卖产生的总收益。但是生成工作负载时的系统时间不同,处理器数量也不同,因此对每小时每个处理器的收益定义如下:

$$\text{收益} = \frac{\text{总处理器数量} \times \text{总小时数}}{\text{总处理器数量} \times \text{总小时数}}$$

类似地,可以定义每个处理器每小时营收及每个处理器每小时成本。

图 1(a)一图 1(c)分别给出了不同负载日志条件下的平均营收、平均成本和平均收益。这些图中的负载按照其正规化负载升序排列。大部分情况下,CAM-DVMPA 算法创造的营收更高。对大于或等于 1.44 的正规化负载,CAM-DVMPA 的营收稳步上升,比 CA-GREEDY 高出 40%。于是得出结论,对资源的需求越高,CAM-DVMPA 创造的营收越高。

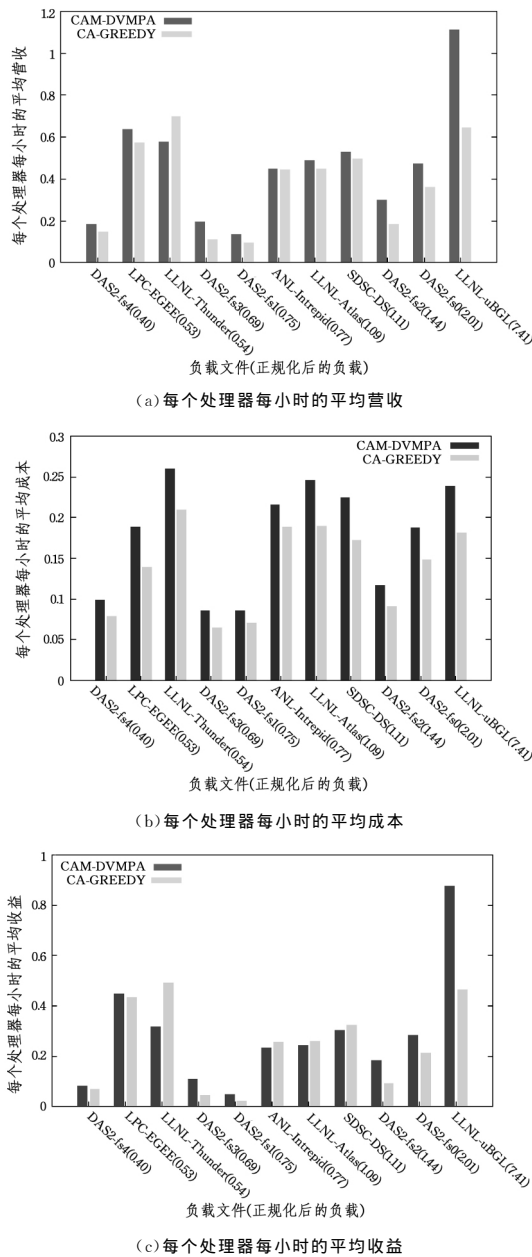


图 1 CAM-DVMPA 和 CA-GREEDY 的性能比较

从图 1(b)中可以发现,无论负载情况如何,CAM-DVMPA 的

总成本均比较高。因为 CAM-DVMPA 是动态确定虚拟机的数量,所以在拍卖时如果申请方相同,则 CAM-DVMPA 分配的处理器数量高于 CA-GREEDY。由于处理器运行时成本较高 ($c_R > c_I$),因此分配的处理器越多,生成的总成本也越高。从图 1(c)中可以看出,当正规化负载大于 1.44 时,CAM-DVMPA 的收益始终高于 CA-GREEDY,且收益间的差异迅速增大,当负载因子低于 1.44 时,CAM-DVMPA 和 CA-GREEDY 分别领先于对方的情况数量相等。这表明当负载较低时,分配机制的效果取决于其他参数。

在图 2 和图 3 中,比较了两种分配算法的资源利用率和被服务用户比例,CAM-DVMPA 在这两方面的指标数值更高。从图 2 可以看到,在大部分情况下,二者的资源利用率相差 30% 左右,即如果从静态分配策略转向动态供应和分配策略,则在这方面可获得显著的性能提升。因为人们已经认可组合拍卖策略在提高分配效率方面的效果,将其与动态供应结合起来即可提高云计算资源分配机制的效率。

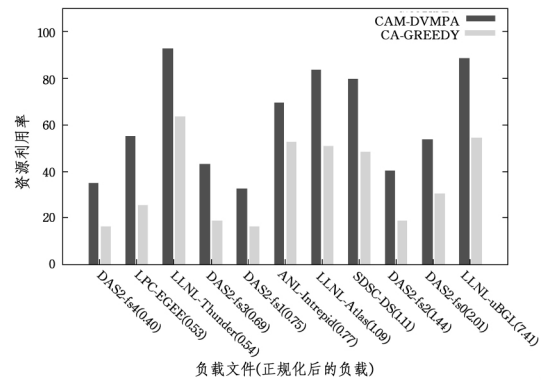


图 2 不同正规化负载条件下 CAM-DVMPA 和 CA-GREEDY 的资源利用率

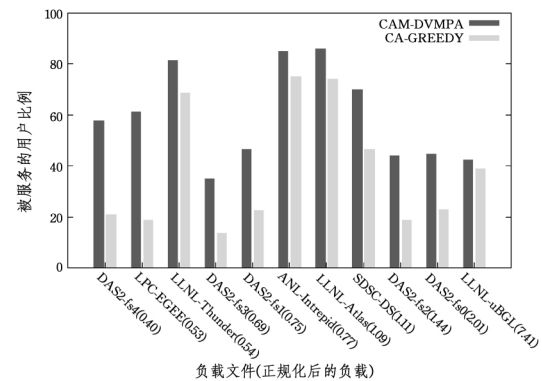


图 3 不同正规化负载条件下 CAM-DVMPA 和 CA-GREEDY 的服务用户比

从图 3 可以看到,CAM-DVMPA 可使更多的用户被服务,因为虚拟机实例不是静态供应。因此,如果无 VM_1 实例可用但是有 VM_2 实例可用,此时若采用 CA-GREEDY 则申请两个 VM_1 实例的用户将申请失败,但是若采用 CAM-DVMPA 则该用户将申请成功。原因在于,CAM-DVMPA 将可用资源看成是两个 VM_1 实例的等价计算资源,因此如果有用户申请两个 VM_1 实例或者申请一个 VM_2 实例,则 CAM-DVMPA 会将可用资源分配给这些用户,具体分配给哪个用户取决于哪个用户的估价更高。这种机制可使 CAM-DVMPA

(下转第 341 页)

结束语 实时的网络安全风险预测能够为管理员采取措施来提供支撑。网络安全风险预测对结果有实时性、灵敏性和准确性的要求,也间接要求模型建立和更新的复杂度不能过高。目前针对网络安全风险预测的研究较多,大多是基于数学建模的方式,对历史数据进行分析处理后进行建模和预测应用。网络安全风险值具有较大波动性,而一些模型并不适用于波动性数据的分析,另一部分模型则具有建模过程复杂的特点。马尔科夫模型既适用于波动性数据的预测分析,又具有建模复杂度可控的特点。因此本文设计了基于马尔科夫模型的网络安全风险预测方法,对日志文件进行分析,使用风险值的方式实现网络风险的量化,采用时变马尔科夫模型对风险值建模并进行预测。使用 DRAPA2000 的数据集进行仿真,实验结果表明该模型具有很好的实时性和较高的准确度,为网络安全风险预测提供了一种新的方式。但是风险值的设定基于经验判断,不同的风险值设定可能会对预测结果产生一定的影响,此处需要进一步开展研究,设定标准,从而高效、准确地对网络进行安全态势预测。

参考文献

[1] 郭祖华,李扬波,徐立新,等.面向云计算的网络安全风险预测模型的研究[J].计算机应用研究,2015(11):3421-3425

(上接第 315 页)

PA 服务更多的 CA-GREEDY 用户。

总的来说,当供需匹配时,CA-GREEDY 的营收更高。如果拍卖期间的资源不像云拍卖那样可配置,CA-GREEDY 拍卖的效率也较高。但是如果资源像云拍卖那样可配置,则难以提前准确预测需求。此时,更应采用 CAM-DVMPA 机制,随着当今技术的发展,该机制可部署为一种无需大量人工干预的独立配置和分配工具。CAM-DVMPA 算法还有另一种用途,可以将 CAM-DVMPA 和 CA-GREEDY 结合起来,周期性地运行 CAM-DVMPA 以确定当前的市场需求,确定与需求最匹配的静态分配策略,然后运行 CA-GREEDY。如果资源利用率低于某一阈值,则调用 CAM-DVMPA 以便再次确定高性能资源配置。这可避免确定 CA-GREEDY 的高效率静态配置时进行详细的统计分析。

结束语 本文研究了云环境下虚拟机实例的动态供应问题,以便在确定基于组件拍卖的虚拟机分配策略时提高收益,并提出 CAM-DVMPA 机制以解决这一问题。利用真实的工作负载数据进行了全面的仿真实验,评估了本文方法的性能。结果表明,CAM-DVMPA 可有效确定市场需求,并针对需求供应计算资源,尤其是在需求较高时可提高云供应商的营收。我们认为本文算法是云环境下 VM 实例分配和供应技术的良好选择。在下一步工作中,我们打算建立一个私有云并在上面部署上述系统。

参考文献

[1] 刘正伟,文中领,张海涛.云计算和云数据管理技术[J].计算机研究与发展,2012,49(1):26-31

[2] 王晋东,沈柳青,王坤,等.网络安全态势预测及其在智能防护中的应用[J].计算机应用,2010,30(6):1480-1482

[3] 陆科达,万励,吴洁明.基于数据挖掘技术的网络安全事件预测研究[J].科技通报,2012,28(6):37-39

[4] 李胜现,田东平,刘建华.基于改进隐马尔可夫模型的网络动态风险评估[J].现代电子技术,2011,34(3):76-77

[5] 黄同庆,庄毅.一种实时网络安全态势预测方法[J].小型微型计算机系统,2014,35(2):303-306

[6] 陈孟婕.电力信息系统动态风险评估技术研究[D].上海:华东理工大学,2015

[7] 刘刚,李千目,刘凤玉,等.面向网络实时风险预测的马尔可夫时变模型[J].兵工学报,2012,33(2):260-261

[8] 刘刚.网络安全风险评估、控制和预测技术研究[D].南京:南京理工大学,2014

[9] MIT Lincoln Lab. 2000 Darpa Intrusion Detection Scenario Specific Data Sets [OL]. <http://www.ll.mit.edu/ideval/data/2000data.html>

[10] 董静.改进的 HMM 网络安全风险评估方法研究[D].武汉:华中科技大学,2008

[11] 雷杰.网络安全威胁与态势评估方法研究[D].武汉:华中科技大学,2008

[2] 张丽敏.云计算中一种高效的虚拟机在线动态分配算法[J].电信科学,2015,31(4):14-19

[3] 黄莉,丁一,姚锦元,等.云采购平台虚拟供应商资源动态分配[J].计算机应用,2014,34(2):377-381

[4] 丁晓波,马中,戴新发,等.一种基于资源预分配的虚拟机软实时调度方法[J].计算机工程与科学,2015,37(5):865-872

[5] 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

[6] Krakov D, Feitelson DG. High-resolution analysis of parallel job workloads[C]//Job Scheduling Strategies for Parallel Processing. Springer Berlin Heidelberg, 2013: 178-195

[7] 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

[8] Roughgarden T. Algorithmic game theory [J]. Communications of the ACM, 2010, 53(7): 78-86

[9] 贲飞,汪芸.云计算下基于容错 QoS 的虚拟机资源分配策略[J].微电子学与计算机,2013,12(3):33-35

[10] 谢文静,唐卓,杨柳,等.基于随机规划的云计算中虚拟机分配优化研究[J].计算机工程与科学,2012,34(5):95-100

[11] Chaisiri S, Lee B S, Niyato D. Optimization of resource provisioning cost in cloud computing [J]. Services Computing, IEEE Transactions on, 2012, 5(2): 164-177

[12] Peng Z, Xu B, Gates A M, et al. The feasibility and properties of dividing virtual machine resources using the virtual machine cluster as the unit in cloud computing[J]. KSII Transactions on Internet and Information Systems (TIIS), 2015, 9(7): 2649-2666