

云计算环境下基于并行计算熵的负载均衡算法

丁 慧

(常州信息职业技术学院, 江苏 常州 213164)

摘要: 针对云计算环境下大量并行任务运行所导致的某些节点负载过重, 从而引起整个系统负载不均和效率低下的问题, 提出了一种基于并行计算熵的资源负载均衡算法; 首先, 描述了云计算虚拟机部署原理并给出了适合云计算环境和异构集群的并行计算熵的计算方式, 然后, 定义了当系统并行计算熵低于阈值时迁移的源物理节点、迁移虚拟机和迁移目标物理节点的确定方式; 最后, 定义了基于并行计算熵的负载均衡算法; 采用 CloudSim 云计算仿真工具对文中方法进行仿真实验, 结果表明文中方法较其它方法的平均负载均衡度约低 21.8%, 具有较低的任务平均响应时间、合理的资源利用率和较小的负载均衡度, 具有较大的优越性。

关键词: 云计算; 虚拟机; 负载均衡; 物理节点

Load Balance Algorithm for Cloud Computing Environment Based on Concurrency Computing Shannon

Ding Hui

(School of Software, Changzhou College of Information Technology, Changzhou 213164, China)

Abstract: In order to solve the problems of lots of operating concurrent tasks leading to the problems such as heavy load balance for some nodes, and therefore leading to the unbalance load and low efficiency of the whole system, a algorithm based on concurrency computing Shannon was proposed. Firstly, the deploy principle for virtual machine in Cloud computing environment was given and the computing for concurrency computing Shannon suiting for cloud computing and heteroid machine was given, then the migrating source physical node, migrating virtual machine and goal physical node when the system concurrency computing Shannon were introduced. Finally, the load balance algorithm based on concurrency computing Shannon was defined. Using the cloud computing tool CloudSim to simulate the method in this paper, and it is proved it can effectively realize the node load balance, and compared with the other methods, it has the advantages such as low average response time, round resource usage and less load balance degree, so it has big priority.

Key words: cloud computing; virtual machine; load balance; physical node

0 引言

云计算^[1-2] (cloud computing) 是大规模集群计算、分布式计算、网格计算等多种技术综合发展的产物, 其采用虚拟化技术并通过互联网, 将各种服务和以及这些服务所依赖的软硬件资源连接起来以透明的方式来提供给用户使用^[3-4]。

资源和服务的虚拟化是云计算区别网格计算的一个重要特征^[5]。虚拟化技术即将一台物理主机映射到多台虚拟机上, 并采用虚拟机代替物理主机来提供服务。目前已有云计算虚拟化产品主要有 VMware 公司生产的 VMotion^[6] 和 Xen 公司生产的 Live Migration^[7]。虚拟化技术为云计算提供了很多便利, 大大提高了硬件利用率, 但这些虚拟机规模巨大和动态变化快等特点, 使得云计算环境下的节点负载均衡策略研究已成为云计算领域的一个重点问题^[8]。

现有的云计算负载均衡策略往往只从系统层面对任务和资源进行调度, 具有一定的片面性和局限性。文献 [9] 研究了一种基于用户 QoS 的云负载均衡机制。文献 [10] 提出了一种基于虚拟机迁移的负载均衡方法。文献 [11] 设计了一种能适应动态网络环境的云计算弹性负载均衡机制。

上述工作具有重要意义, 但仅仅考虑 CPU、内存或带宽

触发迁移, 没有对三者进行综合考虑, 因此, 本文在上述工作的基础上, 提出了一种基于并行计算熵的负载均衡机制, 并通过实验证明了文中方法的有效性。

1 虚拟机负载均衡部署原理

文中设计的虚拟机负载均衡部署模型可以描述如图 1 所示:

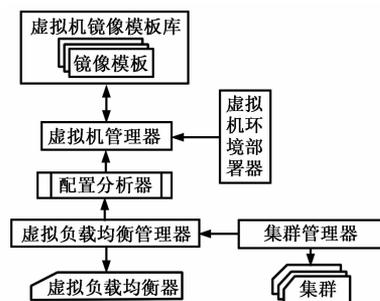


图1 虚拟机负载均衡部署模型

从图 1 可以看出, 文中设计的虚拟机负载均衡部署模型是由虚拟负载均衡管理器、集群管理器、配置分析器和虚拟机管理器组成, 其中各模块的功能如下。

(1) 集群管理器: 对集群物理节点的负载进行监视, 实时计算各物理节点的并行计算熵, 当超过阈值时, 即物理节点发生超载时, 向虚拟机负载均衡管理器发出负载均衡需求, 要求进行负载均衡;

收稿日期: 2013-12-31; 修回日期: 2014-03-01。

作者简介: 丁 慧(1981-), 女, 江苏泰兴人, 硕士, 讲师, 主要从事云计算和数据挖掘方向的研究。

(2) 虚拟负载均衡器：当收到集群管理器的负载均衡请求时，根据节点的并行计算熵选择需要迁移的虚拟器，并将具有最大并行计算熵的节点作为迁移的目标节点，将相关迁移信息发送给配置分析器。

(3) 配置分析器：接收由负载均衡管理器发送的待迁移的源节点、目标节点信息以及其它相关信息；

(4) 虚拟机管理器：根据配置分析器的相关信息，根据虚拟机镜像模板库中虚拟机镜像生成新的虚拟机，并通过虚拟器环境部署器将其在目标节点上部署运行。

2 并行计算熵

并行计算熵是从 shannon 信息熵的基础上发展而来，具有对称性、非负性和扩展性等特征。在迁移过程中的每步都取最大熵增的迁移，能使得所有任务的执行时间最短，因此，当并行运行熵增大时，负载最大物理节点上的总任务计算量趋于减少，而其它物理节点的负载则更为均衡，最终所有任务的总执行时间按减少，所以，负载均衡的目标为：尽可能地增大并行计算熵，以减少所有任务的总执行时间。

并行计算熵的提出是针对同构集群和网络环境，因此，在文中对其进行扩展，将其定义到面向云计算环境和异构集群上。

物理节点 i 的负载 L_i 可以表示为：

$$L_i = \sum_{j=1}^n \omega_{ij} s_j \quad (1)$$

式中， n 为物理节点 i 上运行的所有虚拟机数量总和， ω_{ij} 表示类型为 j 的虚拟机所占的比重， s_j 表示类型为 j 的虚拟机的计算量大小。

物理节点 i 的相对负载 K_i 即某节点负载与所有节点的负载之比，可以表示为：

$$K_i = L_i / \sum_{i=1}^m L_i \quad (2)$$

由于各节点的处理能力即 CPU、内存和带宽具有差异，且这 3 个参数越高，节点的处理性能越好，假设节点的处理能力可以表示为：

$$P_i = \{P_{cpu}, P_{use}, P_{band}\} \quad (3)$$

式中， P_{cpu} 、 P_{use} 和 P_{band} 分别表示可供使用的节点 CPU、内存和带宽，对式 (3) 进行量化并将其表示为一个数值，如下所示：

$$P_{total} = c_1 P_{cpu} + c_2 P_{use} + c_3 P_{band} \quad (4)$$

节点的相对负载可以表示为：

$$K_i' = \frac{L_i / P_{totali}}{\sum_{i=1}^m L_i / P_{totali}} \quad (5)$$

式中， c_1 、 c_2 和 c_3 均为常数，且满足 $c_1 + c_2 + c_3 = 1$ 。

定义 1：当云计算环境下共有 m 个物理节点时，在时刻 t 时，任意节点 i 的相对负载率均可以表示为 $K_i'(t)$ ，则并行计算熵可以定义为：

$$H(t) = \sum_{i=1}^m K_i'(t) \ln\left(\frac{1}{K_i'(t)}\right) \quad (6)$$

3 基于并行计算熵的负载均衡策略

3.1 迁移时间和迁移源节点

集群管理器以采集周期 T_c 收集每个物理节点的所有虚拟机对应的总计算量，并根据式 (6) 计算整个系统对应的并行计算熵并进行判断：

当计算得到的并行计算熵大于或等于阈值 H_m 时，系统负载均衡，各节点均不需要进行虚拟机迁移；

反之，系统负载不均衡，需要进行虚拟机迁移。

迁移源物理节点是负载较重的物理节点，迁移的虚拟机是源物理节点上具有最大计算量的虚拟机，迁移的源物理节点和虚拟机可以通过下述步骤进行确定：

(1) 首先计算系统平均负载：

$$L_{avg} = \sum_{i=1}^m L_i / m \quad (7)$$

(2) 计算各节点相对平均负载的差值：

$$\Delta L_i = L_i - L_{avg} \quad (8)$$

(3) 选取 ΔL_i 大于 0 的节点，并按 ΔL_i 从大到小的顺序将对应的节点放入队列 Q_E 中；

(4) Q_E 即对应了待迁移的源物理节点集， Q_E 中元素的先后顺序对应了待迁移的源物理节点迁移的优先级的先后顺序。

3.2 迁移虚拟机

当确定了需要迁移的源物理节点后，首先将源物理节点上计算量 s_j 大于阈值 s_m 的虚拟机加入到集合 W 中，然后再根据虚拟机的计算量来对 W 排序，根据元素从大到小的顺序将其存在队列 Q_V 中，第一个要迁移的虚拟机即为 Q_V 的队首元素，即为：

$$k = \arg \max_j L_j \quad (9)$$

式中， L_j 表示了待迁移物理节点上的虚拟机计算量的最大值。

3.3 迁移目标节点

迁移的目标节点是整个系统中负载较轻的节点，与获取迁移源节点相似，首先需要计算系统平均负载和各节点相对平均负载的差值，然后选取 ΔL_i 大于 0 的节点，并按 ΔL_i 从小到大的顺序将对应的节点放入队列 Q_I 中， Q_I 中元素的先后顺序对应了目标节点集中虚拟机被迁入的优先级先后顺序。

3.4 负载均衡算法描述

输入：云计算环境下的各物理节点的可用 CPUP_{cpu}、内存 P_{use} 和带宽 P_{band} ，各节点上的所有虚拟机对应的计算量 $S = \{s_1, s_2, \dots, s_m\}$ ；

输出：迁移三元组集 $G = \{ \langle \cup < node_e, v, node_f \rangle \}$ ，即将源物理节点 $node_e$ 上的虚拟机 v 迁移到目标物理节点 $node_f$ 上；

步骤 1：根据公式 (4) 计算各节点性能的量化值，根据式 (5) 计算节点相对负载，然后根据式 (6) 计算系统对应的并行计算熵；

步骤 2：判断系统并行计算熵是否大于阈值 H_m ；

如果大于，则算法结束；

否则，继续执行；

步骤 3：根据 3.1 获取待迁移的源节点集 Q_E ，并取出 Q_E 的队首元素作为当前迁移源节点 $node_e$ ；

步骤 4：根据 3.2 获取待迁移虚拟机队列，并取出 Q_V 的队首元素作为当前要迁移的虚拟机 v ；

步骤 5：根据 3.3 确定迁移目标节点队列 Q_I ，并取出 Q_I 的队首元素作为当前虚拟机迁移的目标节点 $node_f$ ；

步骤 6：将迁移源节点 $node_e$ 、迁移虚拟机 v 和迁移目标节点 $node_f$ 作为三元组加入到集合 G 中，并将即将源物理节点 $node_e$ 上的虚拟机 v 迁移到目标物理节点 $node_f$ 上；

步骤 7：根据式 (4)、(5) 和 (6) 重新更新各节点性能的量化值、节点相对负载和并行计算熵，返回步骤 2 继续迭代。

基于并行计算熵的云计算负载均衡算法流程可以表示为图 2 所示。

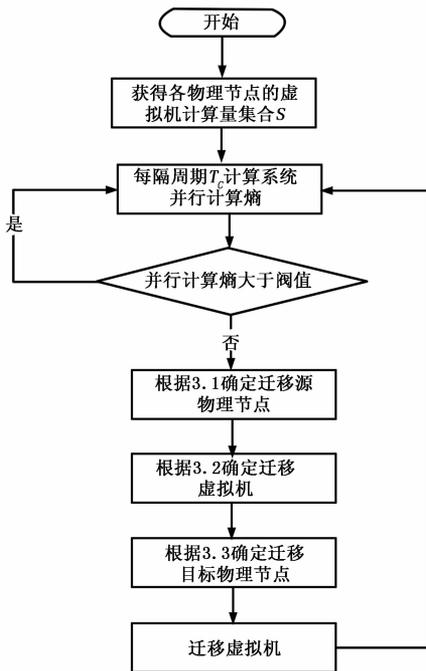


图 2 负载均衡算法描述

4 仿真实验

采用云计

算仿真工具 Cloudsim 对文中方法进行验证，假设某天中任务数量随时间变化的曲线如图 3 所示。

为了衡量文中算法的有效性，从任务平均响应时间、资源利用率和负载平衡度 3 个方面进行评估，并与文献 [11] 进行比较，得到的结果如下所示。

(1) 任务平均响应时间：指集群中的物理节点响应用户任务的速度，可以通过在时刻 t 到来的所有任务的平均完成时间来表示，文中方法和文献 [11] 对应的任务平均响应时间如图 4 所示。

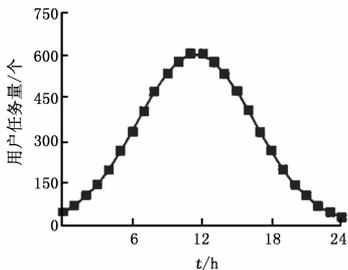


图 3 任务量变化曲线

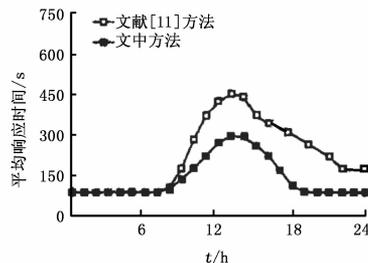


图 4 任务平均响应时间对比

从图 4 中可以看出，文中方法的平均响应时间较少，尤其是在任务高峰时，由于物理节点的均衡使用，所以其平均响应时间低于文献 [11] 约 32%。

(2) 资源利用率：资源利用率可以定义为时刻 t 时，集群中各虚拟机的平均资源使用率，其结果如图 5 所示。

从图 5 中可以看出，文中方法在初期任务较少时，资源利用率较低，而在任务高峰时利用率较高，到末期时由于大部分任务已经处理完，所以资源利用率又减少，而文献 [11] 在整个仿真期间资源利用率都较高，这说明文中方法很好地均衡了

资源负载，能有效地利用现有的资源来处理任务。

(3) 负载均衡度：负载均衡度可以定义为时刻 t 资源使用率方差，其结果如图 6 所示。

从图 6 中可以看出，文中方法的负载均衡度在整个仿真期间均低于文献 [11] 方法，平均约低 21.8%，这说明文中方法的负载均衡程度好。

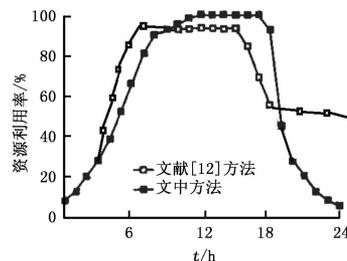


图 5 资源利用率对比

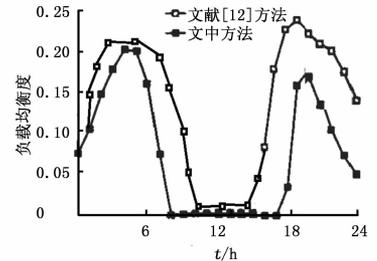


图 6 负载均衡度比较

5 结语

为了实现云计算环境下的节点的负载均衡，设计了一种基于并行计算熵的负载均衡算法。首先，定义了负载均衡部署模型，对各模块进行了介绍和分析，然后描述了负载均衡算法中迁移源物理节点、迁移虚拟机以及目标物理节点的确定方式，并定义了具体的基于并行计算熵的负载均衡算法。实验结果表明文中方法能实现节点的负载均衡，且具有较少的平均响应时间、合理的资源利用率和较低的负载均衡度，具有一定的优越性。

参考文献：

- [1] Cook G, Horn J V. How dirty is your data A look at the energy Choices that Power cloud computing [R]. green peace international technical report, April, 2011.
- [2] 邓 维, 刘方明, 金 海, 等. 云计算数据中心的新能源应用: 研究现状与趋势 [J]. 计算机学报, 2013, 36 (3): 583-596.
- [3] He S J, Guo L, Guo Y K, et al. Elastic application container: A lightweight approach for cloud resource provisioning [A]. In: Proc. of the 2012 IEEE 26th Int' l Conf. on Advanced Information Networking and Applications (AINA) [C], IEEE, 2012: 15-22.
- [4] 马 飞, 刘 峰, 李竹伊. 云计算环境下虚拟机快速实时迁移方法 [J]. 北京邮电大学学报, 2012, 35 (1): 103-106.
- [5] Mauro Andreolini, Sara Casolari, Michele Colajanni. Dynamic load management of virtual machines in a cloud architecture [J]. Department of Information Engineering, 2010 201-204.
- [6] Nelson M, Lim B H, Hutchins G. Fast Transparent Migration for Virtual Machines [A]. Proc. of USENIX' 05 Annual Technical Conference [C]. Berkeley, USA: USENIX Association, 2005.
- [7] Clark C, Fraser K, Hand S, et al. Live Migration of Virtual Machines [A]. Proc. of the 2nd ACM/USENIX Symposium on Networked Systems Design and Implementation [C], Boston, USA: USENIX Association, 2005.
- [8] 王得发, 王丽芳, 蒋泽军. 云计算环境下虚拟机智能迁移技术研究 [J]. 计算机测量与控制, 2012, 20 (5): 1389-1391.
- [9] 叶 枫, 王志坚, 徐新坤, 等. 一种基于 QoS 的云负载均衡机制的研究 [J]. 小型微型计算机系统, 2012, 10 (33): 2147-2152.
- [10] 胡意髓, 欧阳晨, 阙朝坤. 云环境下面向能耗降低的资源负载均衡方法 [J]. 计算机工程, 2012, 38 (5): 53-55.
- [11] 杜 鑫, 郭 涛, 陈俊杰. 云环境下机群弹性负载均衡机制 [J]. 计算机应用, 2013, 33 (3): 830-833.