浙江大学

题目(中)　基于虚拟机间流量亲和性的相关算法研究

姓名与学号　黄昭阳 3130000696

指导教师　金小刚

年级与专业　计算机科学与技术 1302

所在学院　计算机科学与技术学院

摘要

围绕虚拟化集群系统的性能问题，本文在云平台应用场景下假定管理者拥有对于虚拟机之间流量关系和大小的先验知识，将其转化为亲和性的概念构建成图。提出了十分强大的评估虚拟机与物理机匹配程度的函数，提出了使用谱聚类对虚拟机组进行分割的算法。并在这两者基础上深入研究并提出了基于虚拟机流量亲和性的群组初始放置方法，该方法可以使得虚拟化集群系统能够有效减少内部物理机之间的流量数据，并提升后续过程中整个集群系统的稳定性，减少冷/热点机产生的可能性，减少负载均衡维护的频率。

关键词：虚拟机，初始放置，谱聚类，评估函数，亲和性，流量

Abstract

To improve the performance of virtual machine clustering. This document assumes the administrator has the prior knowledge of network flow between the virtual machines, and transforms it into the concept of affinity. Then building a corresponding graph according the affinity. I proposed a strong evaluation function to compute the matching scores between given virtual machines and physical machines, and an algorithm that applying spectral clustering on the virtual machine groups given by the affinity graph to split them. Based on the two previous proposals, I dived into and developed an initially placing method to place the virtual machines connected by affinity into given physical machines based on the network flow affinity. This method can reduce the inner network flow in an virtual machine clustering system, and improve the stability in subsequent processing by reducing the occurrences of cold and hot machines, and the need to load balance.

Keyword: affinity, virtual machine, clustering, network flow, spectral clustering, cloud computing

1. 概述

近年来，云计算发展迅速，越来越多的数据中心提供云计算应用的各种服务。随着云计算的普及和推广，云中的物理资源大量聚集起来。这些物理资源的能耗日益增加，给数据中心成本控制和国家能源供应带来压力。根据美国环境保护署的报告，2011 年数据中心的能源消耗即占到美国电网总量的 2%。而根据 IBM 公司的统计表明，能源成本占数据中心总运营成本的 50%。因此，如何最大限度的利用物理机，为虚拟机管理和分配这些物理资源，成为了云计算服务的一个核心问题。文献[1]把虚拟机放置问题建模成 N 维背包问题，能够有效减少开机物理机的数量，提高资源利用率，但是算法复杂度较高，算法耗时随维度增长迅速，无法保证处理效率。文献[2]将虚拟机放置问题建模成装箱问题（bin packing problem, BPP），通过虚拟机迁移，使用最少的物理机资源满足虚拟机需求，同样达到提高资源利用率的效果。装箱问题已被证明是 NPC 问题，因此许多文献尝试用经典的启发式算法和元启发式算法来解决放置问题。文献[3]采用遗传算法作为解决方案，这种方法能够快速求解，且具有自适应性的特点。但 Falkenauer 和 Delchambre 曾指出[4]，传统遗传算法对于分组问题的适应性很差，并提出了改进的分组遗传算法，用于解决类似装箱问题的分组问题。文献[5]针对虚拟机初始放置问题对分组遗传算法进行了进一步改进，在效率上取得了一定提高。文献[6]基于 CRO 与禁忌搜索混合算法处理虚拟机放置问题，在选择适当参数的情况下收敛速度快于遗传算法。文献[7]对虚拟机之间的流量进行研究，把通信频繁、数据流量大的虚拟机组尽量放置在同一台物理机上，达到减少网络流量、降低虚拟机间网络时延的效果，但没有考虑 CPU、内存等硬件资源。文献[8]基于工作负载预测实现了虚拟机整合，在数据中心提供服务的过程中动

态的调整虚拟机和物理机的映射关系，能够减少物理机使用量和虚拟机迁移开销。

2. 问题描述

虚拟机初始放置问题可被建模为装箱问题，定义如下：

给定一个大小为 M 的集合 S=(s1,s2,…,sm)，表示所有物理机资源；给定一个大小为 N 的集合 P=(p1,p2,…,pn),表示所有虚拟机需求。假设每个物理机或虚拟机都有 D 种资源，如 CPU、内存、网络带宽等，则每台物理机可表示为 si=(ci1,ci2,…,cid)，每台虚拟机可表示为 pi=(ri1,ri2,…,rid)，其中 cij、rij 表示第 i 台物理机或虚拟机在第 j 种资源上的拥有量或需求量。

为了解决放置问题，我们需要了解一下名词的含义：

定义温度 T 为资源占用率，物理机过冷时说明资源利用效率很低，当出现了多个冷点物理机时应当对其进行整合以增加资源利用效率；当出现了热点物理机时应将其分成多个非热点物理机以保证机器运行的稳定性。按照基本的资源分配中不能过热和过冷的要求，给出分配的温度上限 HOT、温度下限 COLD。

热点物理机定义：一台物理机任意维度的资源温度高于 HOT，则该物理机被认为是热点机。

冷点物理机定义：一台物理机任意维度的温度都低于 COLD 时，该物理机被认为是冷点机。

通常，我们认为使用资源与总资源之比为 0.8 为 HOT，使用资源与总资源之比为 0.2 为 COLD，比例系数可以根据实际情况进行调整。

从资源利用率的角度考虑，虚拟机放置问题的目标是在尽量保证虚拟机性能

的情况下采用尽可能少的物理机来承载全部虚拟机需求，以达到提高资源利用率，降低能耗、减少开销的目的，因此我们应当尽量减少冷点机的存在；从系统安全性和可靠性层面考虑，物理机资源应当存在一定程度上的冗余，保证机器不会因为需求峰值的产生而出现卡顿或者宕机的现象，因此我们应当尽量减少甚至避免热点机的存在；从计算机体系结构的层面考虑，当一台物理机上的虚拟机数量过多时，会大大降低计算机指令时间局部性和空间局部性的特点，从而使得计算机整体性能下降，因此我们应当保证一台物理机上的虚拟机数目不会太多；综上所述，所求目标描述如下：

Min($\sum_{i=0}^{m-1} pi$)， pi = $\begin{cases} 0 \ server\ is\ on \\ 1\ server\ if\ off \end{cases}$

所有约束描述如下：

(1)确保每台物理机上所有虚拟机的资源需求总和不会超出物理机的上限，并且不构成热点机；

(2)确保每台虚拟机只被分配到一台物理机上。

(3)确保物理机上的虚拟机数量分配较为均匀，不会在某一台物理机上产生虚拟机拥挤的情况。

3. 资源使用率评估函数

对于每一对即将匹配的物理机 Si 和虚拟机 Pj，需要一个评估函数来确定 Pj 在 Si 上所占资源的比率。本文将该函数称之为 vol 函数。

3.1. 原始评估函数

首先我们从直观上采用的对虚拟机的单维度资源使用率进行评估的函数对其首尾敏感度呈线性关系，对多维度资源则采取连乘，最后取倒数作为该物理

机-虚拟机配对是否适合的评价，其给出的函数形式如下：

$$\mathrm{VOL} = \frac{1}{\prod_{j=0}^{3}\left(1 - \frac{vm_j.used}{pm_j.total - pm_j.used - pm_j.reserved}\right)}$$

函数表达简单直接，但是这样的表达存在两个较明显的问题：

1) 评估函数在区间两端对值的变化不够敏感，这样使得虚拟机在某一个资源维度上接近于 0 或接近于 1 的使用率不能被很好的反映出来。
2) 用单维度资源评估值连乘来确定多维度评估值也会造成部分资源满负荷而其他资源空载的 vol 与所有资源均半负荷的 vol 相等，而我们往往对于快要被使用完毕的资源更加关心。例如，虚拟机 pa=(3,3)在物理机 sa=(10,10)上的评估值 vol=1/(0.3*0.3)=1/0.09；虚拟机 pb=(1,9)在物理机 sa=(10,10)上的评估值 vol=1/(0.1*0.9)=1/0.09。两者的评估值相等，但 pa 明显优于 pb。
3) 评估函数只考虑到当前虚拟机占用资源与元物理机剩余资源之比，并未考虑到物理机的资源总量，而在热点机与冷电机的定义中，我们使用的是使用量与总资源之比，评估函数与需求不吻合，很可能发生使用此评估函数完成初始放置之后出现大量的热点机需要分割，损耗调度系统性能。

3.2. 改进的评估函数

实际上，我们总是不希望虚拟机分配到物理机后使得某一维度的资源几乎为空或几乎为满。前者会造成资源浪费，后者会迫使系统调用热点消除，增添额外的工作。因此评估函数应当在(0,1)区间的边缘更为敏感增长迅速，而在中间稍稍平滑，这样既保持了一定的灵活性，又能避免 3.1.中所述问题。

Sigmoid 函数（如 Figure 1 所示） 是一个良好的阈值函数，能将(-∞,+∞)范围内的值映射到(0,1)，并且中段敏感，两端平滑。Sigmoid 函数的反函数（如 Figure 2 所示）可以将(0,1)范围内的值映射到(-∞,+∞)，并且两端敏感，中段平滑，在其基础上加入权重因子 k 的影响来调整其对低温和高温敏感程度的区别。我们对其取绝对值（如 Figure 3 所示）即可取得理想的单维度资源评估函数。

$$\mathrm{Sigmoid(x)} = \frac{1}{1+e^{-x}}$$

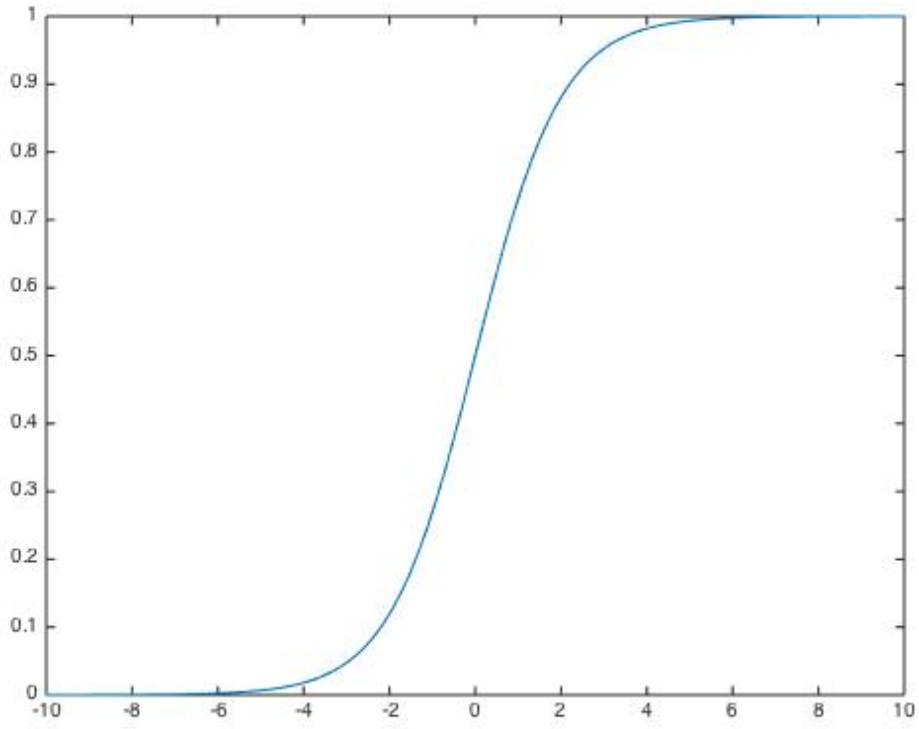

图 1Sigmoid 函数图像

$$Sigmoid^{-1}(x) = \ln\frac{1-x}{x}$$

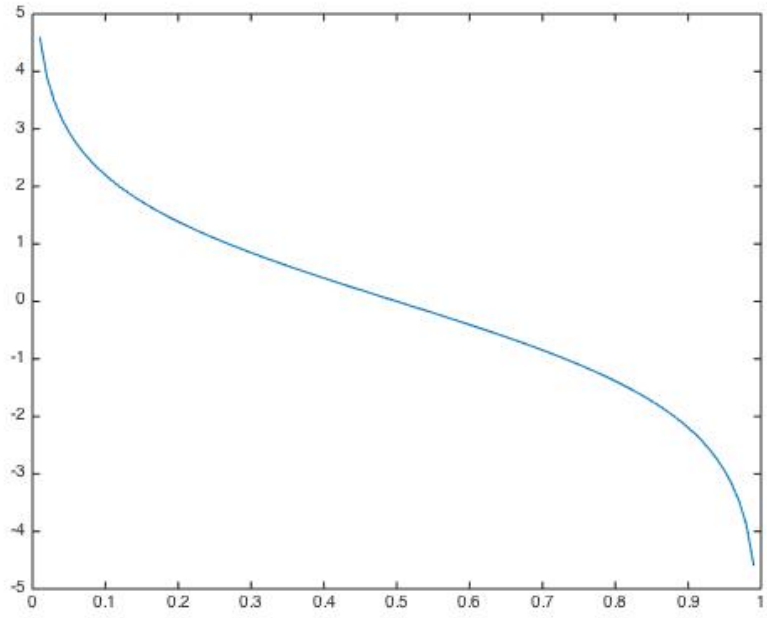

图 2Sigmoid 函数的反函数图像

$$S(x,k) = \left| ln \frac{1-x^k}{x} \right|$$

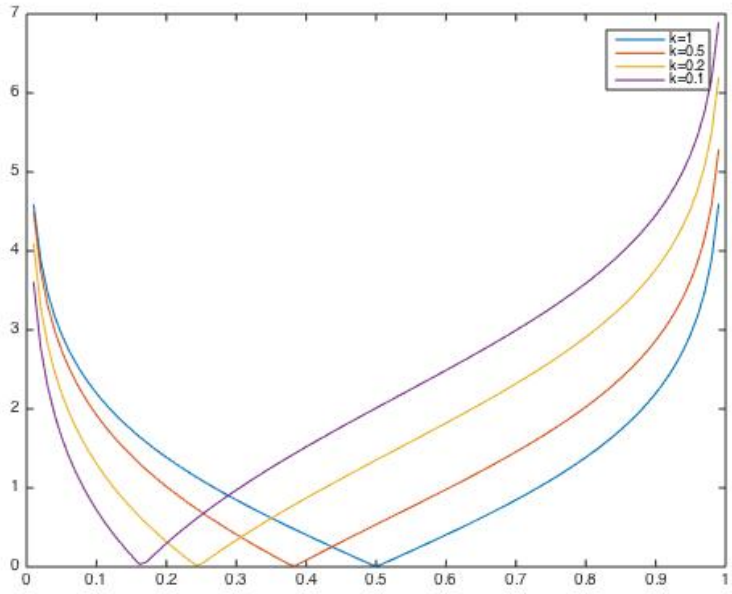

图 3k=0.1, k=0.2,k=0.5,k=1 情况下的损失函数

这个函数能够很好的满足我们对于资源使用率评估函数的要求，并且可以根据对冷热的需要程度调整 k 值，使算法结果倾向于产生更高或更低使用率的物

理机，在初始放置过程中，我们往往宁愿产生冷点也不愿产生热点，并且所有被使用的物理机节点都是从冷点演变过来的，我们往往将 k 值调小使其倾向于产生温度较低的物理机。

在对物理机资源进行评估的时候，我们最关注的应当是那些使用量最拥挤维度的得分，对于使用量较小的维度得分我们往往不是很关注，原评估函数采取的方式是对各个维度的评估函数求积来获取整体的资源评估函数，这种措施有一个缺陷就是较高的得分很容易被较低得分之积所影响。因此我们对其进行调整，使用各个维度资源得分之和获得对物理机资源的评估，这样可以很好的避免高分被多个低分所影响，如 Table 1 所示，我们选取得分较小的物理机来放置，在实际安排中，我们倾向于选择物理机 S1 作为我们放置虚拟机组的候选项，因为物理机 S2 的 CPU 资源使用十分紧张，而物理机 S1 的各个维度资源都属于能够承受的范围内。当我们使用求积函数时我们会选择物理机 S2，而当我们使用求和函数时我们会选择物理机 S1。因此我们最终的出的评估函数如下：

$$\mathrm{VOL} = \sum_{j=0}^{3} S\left(\frac{pm_j.used + pm_j.reserved + vm_j used.}{pm_j.total}, k_j\right)$$

表 1 VOL 得分表

	CPU	Disk	Mem	Eth0	求和	求积
S1	10	10	10	10	40	10000
S2	1000	1	1	2	1002	2000

综上所述，通过我们经过我们改进后的评估函数具有以下优点：

1） 在靠近 0、1 两端的增长十分迅速，使得依据该评估函数进行初始放置后，得到的物理机资源利用率不会过高或过低。

2） 评估依据多个维度的使用率得到总得分对于不均匀的利用比例会产生很好的一致作用，使得依据该评估进行初始放置后，物理机不同维度的资源利用率向接近。

3） 能够很好的与冷电机、热点机的模型定义相吻合，使得依据该评估函数进行初始放置后，较少出现冷、热机器。

4） 能够根据调节系数 k 来调整初始放置后物理机群各个维度的平均温度，根据不同的业务场景进行调整，十分灵活。

4. 基于流量关系的虚拟机分组

在放置虚拟机时，除了考虑虚拟机对物理机的资源利用，还要考虑虚拟机与虚拟机之间的流量关系。由于虚拟化技术的特性，同一物理机上的虚拟机间进行网络通信的延时几乎为零，远小于不同物理机上的虚拟机。因此，尽量将相互之间数据流量较大的虚拟机分在同一物理机上，可以有效减少整个系统的网络负载，降低网络延迟，提高整体效率。

从直观上我们得到一个初步的算法，首先由输入数据得到一张所有虚拟机间流量的网络图，将其中所有的联通子图视为一组有流量关系的虚拟机组。然后，尝试将每组虚拟机都放入单独的一台物理机。若某组虚拟机所需资源超过上限，无法放入物理机，则采用最小割算法将该组虚拟机分割成相互之间流量最小的两个子组，再次尝试放置。递归调用上述分割步骤后，最终所有虚拟机都能全部放置在物理机中，且尽可能地使虚拟机间的流量关系处于同组内，也即同一物理机上。其流程图如下：

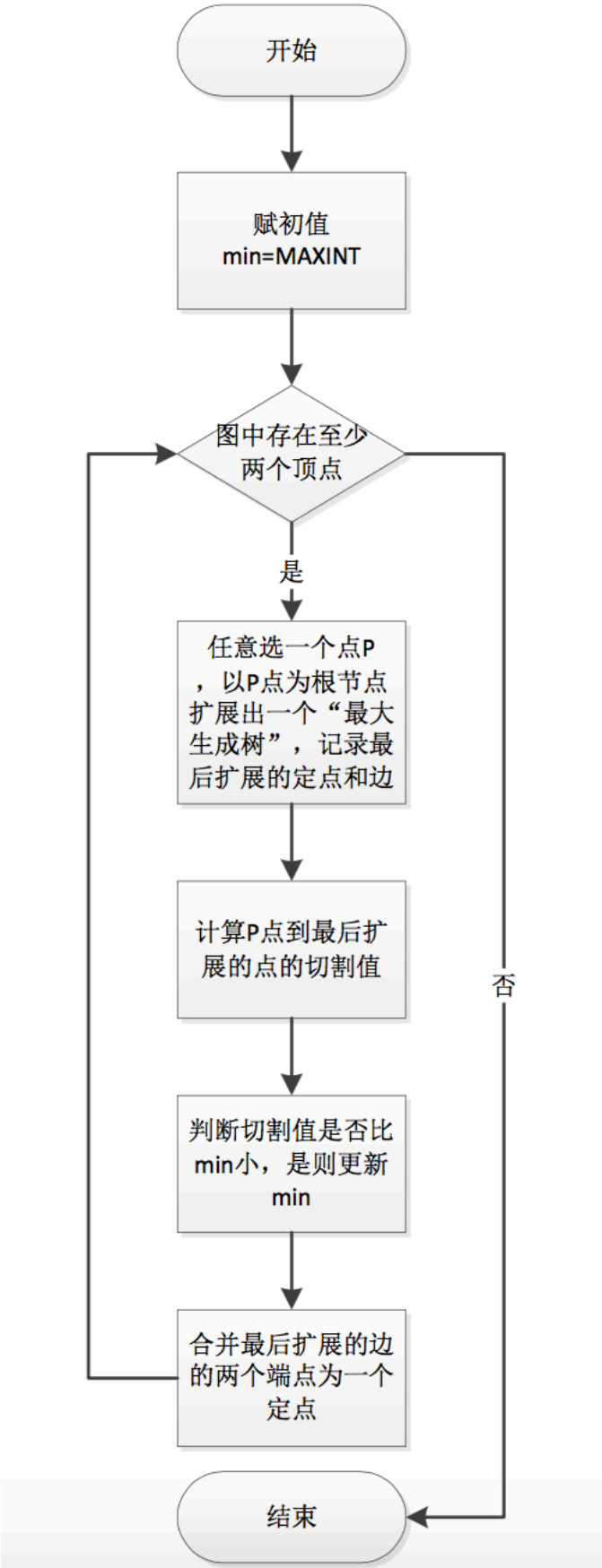

图 4 二分切割初始放置流程图

但是，该算法存在一些比较明显的缺点：

1) 每次分割都只能将一组虚拟机分成两个子组，对于某些远大于物理机容量的虚拟机组，就需要多次递归调用最小割算法，造成系统反应时间变慢。如图 4 所示，该虚拟机组明显至少需分割成 N 组才能放入物理机，而直接使用最小割算法需调用 $\log_2 N$ 次最小割函数。

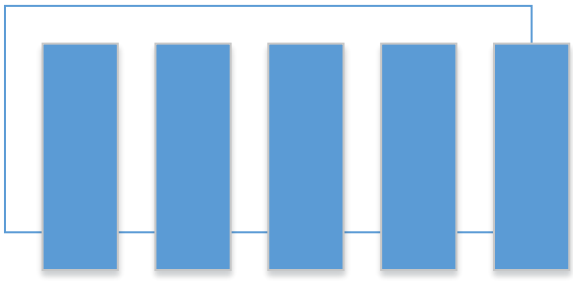

图 5 需要分割成 N 组的虚拟机组

2) 每次固定将虚拟机组分割成 2 个子组不仅效率不高，而且很有可能会进行不必要的分割，从而错过较优解。

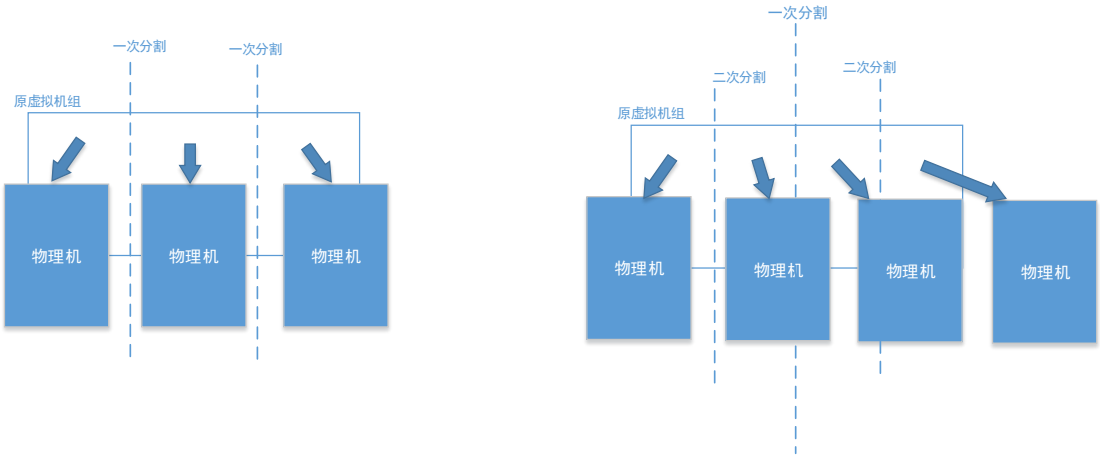

图 6 不必要的分割示意图

3) 直接使用最小割会忽略一个问题，就是子组内虚拟机个数，当一个子组内虚拟机个数较多时，计算机指令的时间局部性和空间局部性特点会被严重干扰，从而较大的降低 CPU 性能，将较多的时间花费在 I/O 上

我们对最小割算法做了较大的改进。首先，得到 N 个虚拟机之后在进行最小割前简单预估需要割成多少个子组，公式为 $K=\max_j\left\{\left\lfloor\frac{group_j}{max\{pm_j\}}\right\rfloor\right\}$ 。得到 K 值后，采用机器学习中谱聚类的方法，构建基尔霍夫矩阵来求最小比例割，其作用在二分割上的优化目标函数是：

$$arg\max_{S,T}\text{cut}(S,T)*\left(\frac{1}{n_1}+\frac{1}{n_2}\right)$$

使用谱聚类的方法对虚拟机进行分割的基础步骤如下：

Step 1：获得 N 个虚拟机及其之间的流量关系，构成一个邻接矩阵

Step 2：构建其对应的基尔霍夫矩阵

Step 3：对基尔霍夫矩阵求其除去 0 特征值外前 K 小的特征值对应的 N 维特征向量，转置后对于 N 个虚拟机我们都获得一个 K 维列向量的描述子

Step 4：使用 K-means 的方法进行聚类得到 K 个子组

最终得到的 K 个分组为上述最优化函数的一个较优解，即考虑组间分割边和划分平衡，这样我们获得的虚拟机分组其数量分布较为均匀。如图 7 所示，在两种分割得到相同的值的情况下，我们倾向于选择 Cut2，因为这种分割方案虚拟机的分布较为均匀 。并且如图 8 所示，我们最终的需求目标也不是纯粹的最小割，而是需要保证虚拟机数目分布较为均衡的尽量小的割，这种需求也与最小比例割的优化目标相吻合。

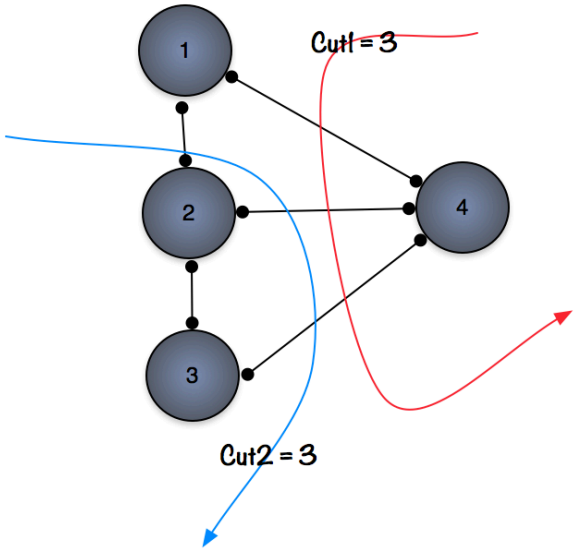

图 7 两种分割方法示意图

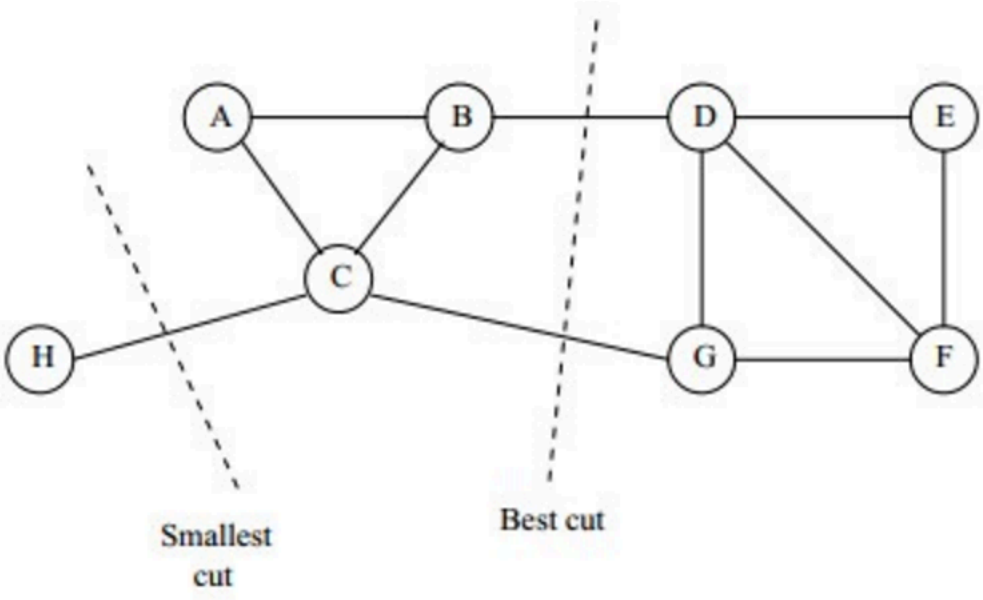

图 8 最小比例割与最小割结果对比

综上所述，我们提出来的分割算法具有以下优点：

1） 能够直接求出 K 分割而不是 2 分割，在虚拟机联通块中点较多时，使用 K 分割能够比 2 分割得到明显更好的时间效率以及更优的解，并且在此算法的基础上，考虑到实际应用，我们获得的虚拟机组邻接矩阵较大时往

往是一个稀疏矩阵，可以利用 k-means 的可并行性和 GPU 矩阵运算加速基尔霍夫矩阵求解的速度。

2） 现代计算机指令集在其空间局部性和时间局部性越明显的情况下，计算机的运行效率更高，我们在放置虚拟机的时候应当考虑如 Figure 6 中的 Cut2 不让他们在物理机上发生拥挤，因此我们的优化目标函数不再是纯粹的最小割，而是最小比例割，这对于提升物理机的运行效率以及减少后期负载均衡的工作很有帮助。

5. 最佳适应（BF）装箱

最佳适应算法是一个常见的用于装箱问题的近似算法。当第 i 个物品需要装箱时，检查所有非空箱子，找到最适合放入该物品的箱子装入。如果不存在可装入的箱子，则开启新的空箱。最佳适应算法倾向于尽量保存较大的空箱以备响应将来可能发生的较大的虚拟机资源请求。

利用该算法，使用资源使用率评估函数判断物品装入某箱子的合适程度，即可在较短时间内得到一个近似解。整体算法流程如下：

1). 读入虚拟机和物理机资源信息

2). 读入虚拟机间数据流量的网络图

3). 求出该网络图的联通子图，将每个联通子图作为一个虚拟机组，加入待放置队列

4). 取出待放置队列中的队头虚拟机组，尝试装箱；若队列为空，跳至步骤 10).

5). 寻找一个能够放下该虚拟机组且 VOL 值最小且不超过阈值 HOT 的非空物理机

6). 若能找到，则将虚拟机组放入该物理机，继续步骤 4).

7). 若找不到，则尝试开启新的物理机；

8). 若存在可符合要求的物理机，则开启该物理机并放入，继续步骤 4).

9). 否则预估一个 K 值，调用谱聚类算法实现最小比例分割将原虚拟机组分割成 K 个虚拟机组，将分割后的子组加入待放置队列，继续步骤 4).

10). 所有虚拟机组放置完毕，算法结束。

6. 小结

这篇论文中我先基于直观给出了一个评估函数以及一种分割虚拟机组的方法，然后根据我们的目标特性进行优化改进，提出了一种新的评估函数，以及使用谱聚类的虚拟机组分割方法，进一步完善了其基于虚拟机流量亲和性系统调度的初始放置算法，我的方法具有以下优点：

1) 评估函数与冷热点评测的吻合度高，放置完成后物理机各维度资源温度相对吻合，不会出现初始放置完成后马上出现热点，较少出现冷点，降低了后续过程中热点、冷点产生的概率，并且能够通过调整系数 k 来调整初始放置后整个系统的平均温度，灵活性高，有利于整个系统的稳定性，降低了之后系统维护的开销。
2) 使用谱聚类求最小比例分割的方法，能够保证虚拟机数目在各组之中分布较为均匀，尽可能的保证了计算机指令时间局部性和空间局部性的特征，较高的提升了物理机的使用效率，提升了整个系统的稳定性。
3) 基尔霍夫矩阵求最小比例割的算法能够直接将原虚拟机组 k 分割，这在虚拟机组较为庞大的时候能够对分割算法有很大的提升，并且由于虚拟机流量邻接矩阵往往是稀疏矩阵的特性，可以使用 GPU 加速，

k-means 算法可以使用并行算法进行加速，算法整体速度将比有较大提升。

[1]Steven S S. The Algorithm Design Manual[M]. New York, USA: Springer, 2010.

[2]Boaz Rafaely. Analysis and design of spherical microphone arrays, IEEE Trans Speech Audio Processing, 2005.

[3] Xu Jing, José A B Fortes. Multi-objective virtual machine placement in virtualized data center environments, 2010.

[4] Falkenauer E, Delchambre A. A genetic algorithm for bin packing and line balancing, 1992.

[5] LI Jin-chao, CHEN Jing-yi. Virtual machine placement research based on improved grouping genetic algorithm, Computer Engineering and Design, 2012

[6] YUAN Xiang, HU Zhigang. Hybrid Algorithm Based On Chemical Reactive Optimization and Tabu Search for VM Consolidation in Cloud Computing Environment, 2013

[7]Vasileios P, Zhang Li. Improving the Scalability of Data Center Networks with Traffic-aware Virtual Machine[C]//Proc. of IEEE INFOCOM'10. San Diego, USA: IEEE Press, 2010.

[8] Wei Liang, Huang Tao. Workload Prediction-based Algorithm for Consolidation of Virtual Machines, Journal of Electronics & Information Technology, 2013