doi:10.3969/j.issn.1000-436x.2014.01.009

IaaS 环境下改进能源效率和网络性能的虚拟机放置方法

董健康，王洪波，李阳阳，程时端
（北京邮电大学 网络与交换技术国家重点实验室，北京 100876）

摘 要：现在的虚拟机放置研究大多集中在物理服务器能源能耗或网络设备能耗的优化，然而随着这些资源的过度聚合，有可能会带来应用性能的下降。提出了一种虚拟机放置方案，主要有 2 个目的：最小化激活物理机和网络设备的个数来减少数据中心能源消耗；最小化最大链路利用率来改善网络性能。此方案在优化网络性能的同时，减少物理服务器和网络设备的能耗，使得能源效率与网络性能达到平衡。设计了一种新的二阶段启发式算法来求解，首先，利用基于最小割的层次聚类算法与最佳适应算法相结合来优化能源效率，然后，利用局部搜索算法再次优化虚拟机位置来最小化最大链路利用率。仿真实验结果表明，所提方案取得了良好的效果。

关键词：IaaS；虚拟机放置；网络性能；能源效率
中图分类号：TP301 **文献标识码**：A **文章编号**：1000-436X(2014)01-0072-10

Improving energy efficiency and network performance in IaaS cloud with virtual machine placement

DONG Jian-kang, WANG Hong-bo, LI Yang-yang, CHENG Shi-duan
(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: The existing virtual machine(VM) placement schemes mostly reduce energy consumption by optimizing utilization of physical server or network element. However, the aggressive consolidation of these resources may lead to network performance degradation. In view of this, a VM placement scheme was proposed to achieve two objectives. One is to minimize the number of activating physical machines and network elements to reduce the energy consumption, and the other is to minimize the maximum link utilization to improve the network performance. This scheme is able to reduce the energy consumption caused by physical servers and network equipment while optimizing the network performance, making a trade off between energy efficiency and network performance. A novel two-stage heuristic algorithm for a solution was designed. Firstly, the hierarchical clustering algorithm based on minimum cut and best fit algorithm was used to optimize energy efficiency, and then, local search algorithm was used to minimize the maximum link utilization. The simulations show that this solution achieves good results.

Key words: IaaS; virtual machine placement; network performance; energy efficient

1 引言

云计算数据中心大多采用虚拟化技术，租客订购一组相互之间通信、放置在不同主机上的虚拟机，并与云提供者签订 SLA(service level agreement)。每个虚拟机需要一定数量的资源，如 CPU、内存、磁盘、带宽等，能维护应用之间的性能隔离和安全。同时，虚拟化技术可在同一台物理机上运行多台虚拟服务器[1]，提供资源整合和复用，有利于提高资源利用率并节约设备能耗。这样，虚拟化

收稿日期：2013-09-07；**修回日期**：2013-12-16
基金项目：国家自然科学基金资助项目（61002011）；国家高技术研究发展计划（“863”计划）基金资助项目（2013AA013303）；中央高校基本科研业务费专项基金资助项目（2013RC1104）；软件开发环境国家重点实验室开放课题基金资助项目（SKLSDE-2009KF-2-08）
Foundation Items: The National Natural Science Foundation of China (61002011); The National High Technology Research and Development Program of China (863 Program) (2013AA013303); The Fundamental Research Funds for the Central Universities (2013RC1104); The Open Fund of the State Key Laboratory of Software Development Environment (SKLSDE- 2009KF-2-08)

技术也可以帮助管理者实现资源的有序、按需调配，为云计算的资源灵活管理和绿色节能提供了一种有效的解决办法。

对于采用虚拟化技术的公共云(public cloud)而言，其提供的主要服务之一是基础设施即服务(IaaS, infrastructure as a service)，像 Amazon EC2[2]。在 IaaS 中，租客向云提供者付费租用所需的虚拟机资源。云提供者按照租客所提出的虚拟机需求，遵守双方签订的 SLA，利用虚拟机在物理机上的灵活放置，优化数据中心的资源配置。由于不同的虚拟机与物理机的映射关系会带来不同的资源利用率，就云提供者来说，需要关心的一个主要问题是:租客申请的多个虚拟机在云提供者的物理服务器上如何放置，既可以满足应用的性能要求，又能提高资源利用率和减少能源消耗，从而减少数据中心的运营和管理成本。把此类问题定义为虚拟机放置问题，这是当前云计算研究中的热点问题。

对于虚拟机放置问题，一个主要的研究方向是通过资源的整合，最小化激活物理设备(物理服务器、交换机、链路等)的数目，使空闲设备处于休眠状态，从而减少能源消耗。其中，一些研究工作[3~6]利用虚拟机放置，提高物理服务器的利用率，减少应用服务主机的数目，从而减少能源的消耗。但这些研究较少考虑网络资源的优化，并没有考虑网络拓扑和当前网络流量的影响，而网络资源是数据中心的稀缺资源，直接影响到应用的性能[7]；另一些研究[8~10]利用虚拟机放置来提高网络设备的利用率，结合路由的优化，减少网络设备的数目，以此优化网络能源效率。但是，不论是优化物理服务器能源效率，还是优化网络设备的能源效率，都会带来资源的过度聚合，会影响应用的性能，提高了 SLA 的违反率。特别是网络流量的聚集，会造成链路热点，带来网络拥塞问题。

有鉴于此，云提供者要考虑优化物理服务器和网路设备的能耗，整合虚拟机到更少的物理机上，关闭空闲物理机；优化网络总流量，减少流量所用网络设备（如交换机、链路等）的数目，节约网络设备能源；也要考虑资源过度聚合带来的网络拥塞问题。使得能源效率与网络性能达到平衡，在尽量满足应用的 SLA 的条件下，最小化能源消耗。

本文提出了一种多资源条件约束的虚拟机放置方案。在满足物理机多个资源（CPU、Memory 等）和网络链路容量约束的情况下，通过对放置在物理机上的虚拟机进行交叉优化放置，在尽可能避免网络拥塞的前提下，优化物理服务器和网络设备的能耗，使得空闲的资源处于休眠状态，最小化激活物理服务器和网络设备的数目，从而减少数据中心的能源代价。

虚拟机放置对于物理机能源的优化可以抽象为装箱问题[11]，最小化激活的物理服务器数目。而对于网络资源的优化，主要利用网络拓扑和当前的网络流量，可以把这一问题抽象为二次分配问题[12]，最小化网络中的总流量，网络中总的通信流量较小，所激活的网络设备的数目也较少。同时最小化最大链路利用率，避免网络拥塞发生，保证网络性能。既要减少激活物理机和网络设备的数目，来降低数据中心的能源消耗，又要保证应用性能，避免网络拥塞。这是一个经典的多目标优化问题[13]。而装箱问题和二次分配问题是 NP 难问题，本文设计了一种新的二阶段启发式算法求解多目标优化问题。第一阶段，最小割的层次聚类算法与 BF(best fit)算法相结合。利用层次聚类算法，用最小割算法把流量相关的虚拟机聚类在一起，使得流量大的虚拟机放置在同一个物理机上或同一个接入交换机下，来减少网络中的流量。然后，根据聚类结果，利用 BF 放置算法来最小化激活物理机数目。第二阶段，利用局部搜索算法再次优化虚拟机位置，目的是最小化最大链路利用率，从而使得数据中心网络流量分布均衡，减少拥塞链路的产生。算法要能适应物理机与虚拟机大小异构的情况。仿真实验表明，与 BFD (best fit decreasing)算法、随机放置算法相比，取得了良好的效果。

2　相关工作

对于减少能耗的虚拟机放置问题的研究主要集中在 2 类。一类是降低物理服务器能耗[3~6]。文献[3]在虚拟化系统中，动态调整应用在服务器的位置对能源的影响，考虑应用的迁移成本并能预计出迁移后的能源消耗，使用一种简单的算法证明了通过动态迁移技术可以实现数据中心能耗成本的节省。文献[4]采用负载预测技术，在最小化激活物理机的同时不影响应用性能。文献[5]是按照用户设置 max、min、share 等虚拟机设置参数提供一种新的物理机资源分配方法，整合多个虚拟机到物理机，使得被放置物理机的资源利用率高，并且使得物理机的能耗较低。但是，该文只

考虑 CPU 资源进行优化，而没有考虑其他资源的优化。文献[6]是把虚拟机带宽到物理机带宽整合问题看成 NP 难的随机装箱问题，它表明一定尺寸的虚拟机以某种概率分布放置在相应的物理机上，而优化目标是物理机数量最小。这些研究只考虑优化物理服务器的能耗，而较少考虑数据中心其他资源的优化。

另一类是降低网络设备的能耗[8,9]，文献[8]通过虚拟机迁移技术和网络路由的优化来减少数据中心网络能源消耗，从而节约数据中心电源损耗。文献[9]同时优化虚拟机位置和流量路由以尽可能多地关闭空闲网络设备来节省数据中心网络能耗。这2个方案只假设满足物理服务器的资源需求，优化了数据中心网络资源，没有优化物理服务器能耗。

当前，也有一些研究[10, 14~17]利用虚拟机放置技术优化应用性能。文献[10]用流量相关的虚拟机位置来改善数据中心网络的扩展性，通过优化虚拟机在物理主机的位置，使虚拟机的流量与它们的网络物理距离相关，通信流量大的虚拟机对可以放置在距离相近的 2 个物理主机上，从而减少数据中心网络的总体流量。文献[17]考虑了最小化所有服务器与终端之间的网络总负载，提出了一个基于虚拟机迁移的在线算法。文献[10, 17]都没有考虑云数据中心的能源优化。文献[14]利用虚拟机放置致力于提高物理节点的利用率，同时通过调整流量路由优化网络链路利用率。但是它们没有考虑优化数据中心网络设备的能耗。文献[15]提出了一个应用相关的虚拟机分配方案，在满足 SLA 需求的条件下，改善硬件资源的利用率，但没有考虑网络流量的优化。文献[16]提出了一个新的虚拟机动态整合自适应启发式算法，尽管这个算法在确保SLA不被违反的前提下减少了物理机的能耗，但没有考虑网络性能的优化。本文的方案是在保证网络性能的前提下，减少数据中心的物理机和网络设备的能耗。

3 建模

在当前的云计算中，按需分配资源是云计算的主要特征，虚拟化是实现这一特征的主要技术，而 IaaS 是云计算主要的应用之一。云提供者给租客提供虚拟机，对于租客而言，最关键的一个服务就是租用虚拟机要满足 QoS(quality of service)的需求和应用的性能保障。对于云提供者而言，如何设计虚拟机放置方案来提高资源利用率并能改善网络性能是一个重要问题。

现在，虚拟机放置研究主要在能源节省、故障容忍、QoS 管理等方面，也大多集中在服务器资源的优化，没有考虑网络拓扑和当前网络流量的影响，而网络资源是数据中心的稀缺资源，直接影响到应用的性能。而虚拟机放置可以改变虚拟机的位置，即改变其所属的物理服务器，从而改变流量的发送端和接收端，以达到控制和优化数据中心网络流量的目的。本文的虚拟机放置方案主要集中在下面 2 个方面。

1) 能源优化。这里的能耗主要包括两部分：物理服务器和网络设备。通过虚拟机与物理机的不同映射关系最小化激活物理机和网络设备的个数，关闭或休眠空闲的物理设备，减少数据中心的能源消耗。

2) 网络性能优化：通过改变数据中心虚拟机的位置，从而改变了虚拟机之间的流量路径，最小化网络最大链路利用率，达到改善网络性能的目的。

本文的虚拟机放置问题是：从云提供者的角度，根据不同租客的资源需求，在不违反 SLA 的前提下，也就是说，在保证网络性能的前提下，设计一种虚拟机放置策略，提高资源池中的资源利用率，减少激活物理机和网络设备的数量。这样就降低了云计算数据中心的硬件投入和能源消耗，且减少了数据中心的运营成本。

3.1 数据中心能源优化

3.1.1 能源模型

建模数据中心的能耗主要由以下三部分组成：物理机能耗、网络设备能耗、其他能耗。在式(1)中，物理机能耗 E_{ser} 主要由 CPU、内存、存储和网络接口等组成，网络设备能耗 E_{net} 主要包括交换机、链路等。其他能耗 E_{other} 主要包括制冷、湿度控制、照明等，像制冷涉及到数据中心规模、室外温度、周边环境、制冷技术（风能、水能）等多种因素。笔者的目的在于利用虚拟机以及网络流量的整合改进能源效率，关闭/休眠空闲的物理机和网络设备来节约能耗。本文聚焦于物理机和网络设备的能耗，暂不考虑其他能耗。当数据中心物理设备能耗下降时，伴随着发热量的下降，相应地也会减少冷却的能耗，有利于其他能耗的降低。本节主要符号定义及其意义如表 1 所示。

表 1　主要的符号定义及其意义

符号	含义
M	物理机数目，任一物理机 m=1,…,M
N	虚拟机数目，任一虚拟机 i=1,…,N
$\vec{\boldsymbol{H}}_m$	物理机 m 的 d 维资源向量，取值为 $\{H_{m,1},H_{m,2},\cdots,H_{m,d}\}$，$d$ 为资源类型数
$\vec{\boldsymbol{S}}_i$	虚拟机 i 的 d 维资源向量，取值为 $\{S_{i,1},S_{i,2},\cdots,S_{i,d}\}$，$d$ 为资源类型数
Y_m	二进制变量，为 1 表示物理机 m 为激活状态，为 0 表示物理机 m 为休眠状态
$X_{i,m}$	二进制变量，当虚拟机 i 放置在物理机 m 上，为 1；反之，为 0
E_{ser}	物理机的能耗
E_{net}	网络设备的能耗
E_{DC}	数据中心能耗
$Cost_{\text{net}}$	最大链路利用率

$$E_{\text{DC}} = E_{\text{ser}} + E_{\text{net}} + E_{\text{other}} \tag{1}$$

3.1.2　物理机能耗优化

对物理机能耗的优化抽象为多维资源约束的装箱问题。目的是最小化活动物理服务器的数目。

$$\min \quad E_{\text{ser}} = \sum_{m=1}^{M} Y_m \cdot E_{\text{ser}}^m \tag{2-1}$$

$$\text{s.t.} \quad \sum_{i=1}^{N_m} X_{i,m} \cdot \vec{\boldsymbol{S}}_i \leqslant Y_m \cdot \vec{\boldsymbol{H}}_m \tag{2-2}$$

$$\sum_{m=1}^{M} X_{i,m} = 1 \tag{2-3}$$

其中，N_m 表示物理机m上的虚拟机个数。E_{ser}^m 表示物理机 m 的能耗。式(2-2)表示放置在同一台物理机上的多个虚拟机资源容量总和小于物理机容量。式(2-3)表示任一虚拟机只能放置在一台物理机上。

物理机能耗 E_{ser}^m 与其承载的虚拟机数量、虚拟机的负载情况等均有关系。而物理机所承载的虚拟机数量、虚拟机的负载情况都会归结到对物理机负载的影响。所承载的虚拟机数量多，虚拟机本身的负载大，物理机的负载就大。采用式(3)和式(4)来建模物理机能耗 E_{ser}^m。在式(3)中，$P_{\max}$ 是物理机满负载的最大功耗，k 的一般取值为 0.7，也就是说，空闲物理机的功耗达到最大功耗的70%左右。这说明物理机在空负载的情况下，如果不关闭或休眠物理机，物理机的能耗还是很大的。u 是资源利用率，由于现在的物理服务器只有 CPU 采用 DVFS (dynamic voltage and frequency scaling)技术，而其他部件并没有采用这项技术，所以 u 主要是指 CPU 利用率。在这里利用率 u 与功耗采用线性关系。

$$P(u) = kP_{\max} + (1-k)P_{\max}u \tag{3}$$

$$E_{\text{ser}}^m = \int_t P(u(t))\,\mathrm{d}t \tag{4}$$

3.1.3　网络设备能耗优化

对于网络资源的优化，目标是最小化数据中心的通信总流量，通信总流量的最小化如式(5)所示，把此问题抽象为二次分配问题。利用虚拟机放置，把相互之间流量大的虚拟机尽量整合到同一个物理机上，或放在同一个网络交换机下。对于数据中心的对分带宽拓扑，通信总流量小，那么所需活动的网络设备（交换机、链路等）的数目就会减少。使得空闲的网络设备处于休眠状态，从而降低能源的开销。采用这种方案，既可以节省网络设备的能源，也可以节省网络带宽。

$$\min \quad E_{\text{net}} = \sum_{i=1}^{N_{swi}} E_{swi}^i + \sum_{i=1}^{N_{\text{link}}} E_{\text{link}}^i \tag{5}$$

其中，E_{swi}^i 表示交换机 i 的能耗，N_{swi} 表示活动交换机的数目，E_{link}^i 表示链路 i 的功耗，N_{link} 表示活动链路的数目。N_{swi} 和 N_{link} 的计算，利用 Elastictree 方案提出的背包算法[18]，对于每条流，用贪婪背包算法选择一个满足容量的靠左链路。在数据中心层次化拓扑中，每一层的路径选择采用从左至右，而不是采用平均分配流的随机选择策略。当所有的流都被分配后，算法返回一个有流量经过的网络设备子集，而没有流量经过的网络设备可以处于休眠状态或关闭。通过虚拟机放置技术和流路径的结合，使得流量使用较少的网络设备。

3.2　网络性能优化

从流量工程的角度来看，最小化最大链路利用率是网络性能优化的主要目标。$f_{s,t}^{i,j}$ 表示分配到链路(s, t)上虚拟机对(i, j)的流量，$C_{s,t}$ 表示链路(s, t)的容量。

网络链路利用率可用 $l_{s,t}$ 表示为

$$l_{s,t} = \frac{\sum_{(i,j)} f_{s,t}^{i,j}}{C_{s,t}} \tag{6}$$

要使得 $l_{s,t}$ 尽可能小，问题可表示为

$$\min \quad Cost_{\text{net}} = \max\{l_{s,t}\}$$
$$\text{s.t.} \ \sum_j f_{s,t}^{i,j} - f_{t,s}^{i,j} = 0, \ X_{i,s} = 1, \ X_{j,t} = 1$$
$$\sum_{i,j} f_{s,t}^{i,j} \leqslant C_{s,t}$$
$$f_{s,t}^{i,j} \geqslant 0 \tag{7}$$

3.3　总体目标

本文的总体目标是最小化能耗以及最小化最大链路率，可形式化为

$$\min \ \ g = E_{\text{DC}} + r \cdot Cost_{\text{net}} \tag{8}$$

其中，r是正常数，g是能耗与最大链路利用率的加权和。这是经典的多目标优化问题。

4　算法

对于多目标优化和NP难问题，一般求解采用启发式智能算法，如遗传算法、蚁群算法等，但此类算法存在算法时间性能较差、结果不稳定等问题，本文设计了一种新的两阶段启发式算法求解多目标优化问题，首先，在虚拟机放置过程中，在不发生网络拥塞的情况下，以优化能耗为主。提出了一种基于最小割的层次聚类算法与BF算法相结合来求解多目标优化问题。进行层次聚类算法，用最小割算法把相关的虚拟机聚类在一起，最小化网络中的总流量；根据聚类结果，利用BF放置算法来减少物理机的能源消耗。其次，在虚拟机放置过程中，如果有网络拥塞，采用最小化网络最大链路率，以优化性能为主。基本输入为：网路拓扑、链路容量、流量路由、虚拟机之间流量需求、虚拟机需求向量组和物理机向量组。基本的输出为：虚拟机在物理机上的映射关系。

4.1　基于最小割的层次聚类算法

现在数据中心大多是三层体系结构[7]，考虑数据中心网络拓扑特征和当前的虚拟机之间的网络流量，把流量大的虚拟机对尽可能放置在同一个物理机上或同一个交换机下，来减少网络中的流量总量。这样保证了应用的性能，也减少了流量所用网络设备的个数。建模网络流量的优化为二次分配问题。这里，可以考虑采用基于虚拟机流量的层次聚类算法来解决二次分配问题。建模图$G=(V,E)$，其中，V是虚拟机的集合，E是虚拟机之间的流量，利用图G的最小割算法来实现层次聚类。图$G=(V,E)$是一无向图，给定节点集合$Q \subseteq V$，而$\delta(Q)$表示边的集合，这些边的一个顶点在集合Q中，另一个顶点属于$V \backslash Q$。当$Q \neq \varnothing$或$Q \neq V$，$\delta(Q)$中的边就组成一个割集，表示为$(Q, V \backslash Q)$。

每一条边$(i,j) \in E$，有一个非负的容量$C_{i,j}$。而一个割集的容量可以定义为割集中每条边容量的总和。也就是$C(Q, V \backslash Q) = \sum_{i,j \in \delta(Q)} C(i,j)$，最小割问题就是在图$G$中找一个容量最小的割集。

如图 1 所示，把图 G 最小割的结果用二叉树$T(V)$表示，建一个二叉树 $T(V)$，左子树 TL 为 Q 里的节点，权重为Q中边值的和，$W(\text{TL}) = \sum_{i,j \in Q} C(i,j)$；右子树 TR 为 V/Q 的节点，权重为 V/Q 中边值的总和，$W(\text{TR}) = \sum_{i,j \in V \backslash Q} C(i,j)$，如果 $W(\text{TL}) < W(\text{TR})$，则把左子树 TL 与右子树 TR 互换，意味着左子树 TL 里虚拟机通信流量一直大于右子树。二叉树 $T(V)$的叶子节点当且仅当表示一个虚拟机时，树枝为经过聚类的虚拟机集合。把这个算法称为 MC-BT 算法，算法描述如图 2 所示。

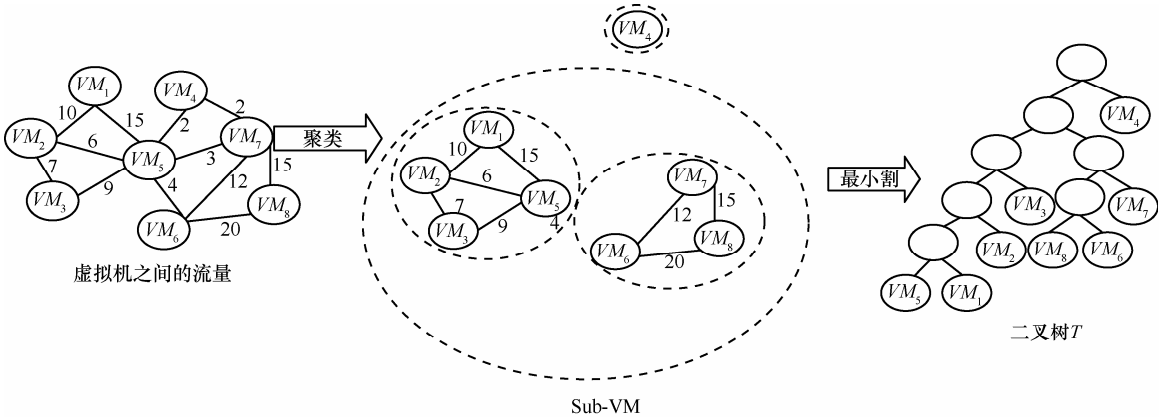


图 1　基于最小割的层次聚类算法

```
1)  输入: A（流量矩阵）
2)  输出: T(V)（按流量聚合的流量二叉树）
3)  Initial cut S
4)  Initial Binary Tree T
5)  Matrix A 转换为 Graph G=(V,E)
6)  While G 中有超过一个节点 do
7)    选取 2 个节点 s 和 t
8)    计算最小割容量 δ (S')
9)    分割 s 和 t
10)   If c(S',V\S')<C
11)     C←C(S', V\S') and S←S';
12)   Endif
13)   左子树 TL←Gs(V), 计算 W(TL)
14)   右子树 TR←Gt(V), 计算 W(TR)
15)   If  W(TL)<W(TR)
16)     TL 与 TR 互换，Gs 互换 Gt
17)   End if
18)   替换 G by Gs and Gt
19) Endwhile
20) Output T
```

图 2　基于最小割的层次聚类算法

4.2　基于 BF 的虚拟机放置算法

对于MC-BF得到的BT(V)，先序遍历树中的所有叶子节点放在向量$\boldsymbol{VM}_{\text{list}}$里，$\boldsymbol{VM}_{\text{list}}$里节点就是要放置的所有虚拟机节点，从前面的论述中可以看出：1) 在$\boldsymbol{VM}_{\text{list}}$里，排在前面的节点一般是流量较大的虚拟机；2) 在$\boldsymbol{VM}_{\text{list}}$里，每个节点(虚拟机)的前后邻居就是跟它通信流量的虚拟机，与当前节点的位置距离越远，说明它们之间的通信流量越小。

如图3所示，把$\boldsymbol{VM}_{\text{list}}$里大小不一的虚拟机节点放置到对应的物理机上，采用最佳适应算法。从$\boldsymbol{VM}_{\text{list}}$中按序开始依次放置虚拟机。当放置某个新到来的虚拟机时，从已使用的第一台物理机开始依次搜索，找到一台与虚拟机大小最匹配的物理机进行放置，只有当所有的物理机都不能容纳这个虚拟机时，启用一台新的物理机。将这个算法称为结合流量层次聚类的最佳适应（BF-HC）算法。算法描述如图4所示。

```
1)  输入: PMlist(物理机的资源向量组)，T(流量二叉树)
2)  输出: X(虚拟机映射物理机的矩阵)
3)  Initial VMlist
4)  先序遍历树 T 中的所有叶子节点依次放入 VMlist 中
5)  For each VMi in VMlist do
6)    Foreach PMm in PMlist do
7)      If (isAllocable(VMi,PMm) and
8)          PMm.spare < PMbest.spare
9)          Best←m
10)     Endif
11)   Endfor
12)   VMi,best ←1
13)   Allocation (VMi,PMbest)
14) Endfor
15) Output X
```

图 4　结合流量层次聚类的最佳适应算法

4.3　局部搜索算法

如当前网络不发生拥塞，只采用BF-HC算法优化云计算数据中心的资源能耗。当网络出现热点链路时，在BF-HC算法的基础上，则只采用局部搜索算法优化链路利用率，避免拥塞的出现。本文称这个算法为BF-HC-LS算法，算法描述如图5所示。选择产生拥塞链路的流量最大的虚拟机，随机与左右邻居

```
1)  输入: X（虚拟机与物理机映射矩阵）、A（流量矩阵）、TP
（网络拓扑）、Nmax（最大迭代数）
2)  输出: Xbest
3)  Initial  X, A, TP
4)  While s=1 to Nmax do
5)    选择拥塞链路下，流量大的虚拟机
6)    与左右邻居交换机下虚拟交换
7)    得到新的 X'
8)    计算 MLU'
9)    If MLU'<MLUbest
10)     X←X' and  MLU←MLU'
11)   Else
12)     以一定概率接受 X'、MLU'
13)   Endif
14) Endwhile
15) Output Xbest
```

图 5　局部搜索优化最大链路利用率

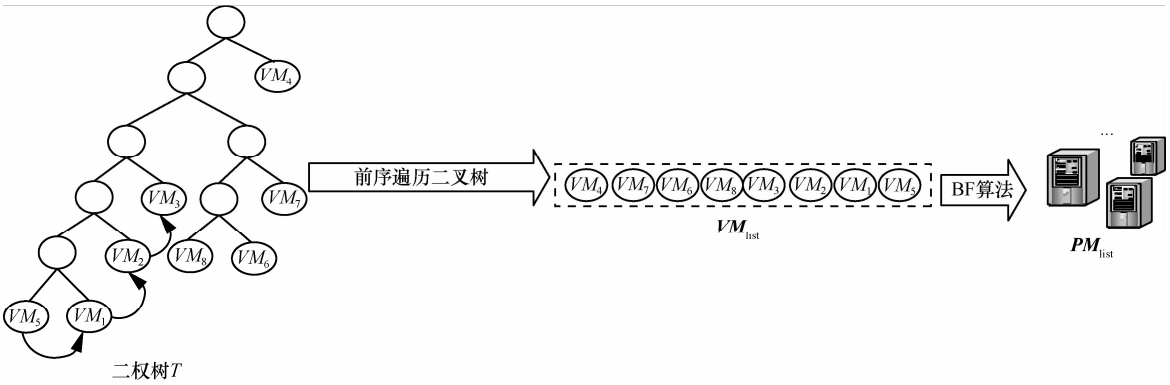


图 3　基于 BF 的虚拟机放置算法

交换机下的虚拟机交换，计算目标函数：最大链路利用率或热点链路数目。如果目标函数值减小，则接受此次交换，如没有减少，也可按一定概率接受。依次重复，直至循环到设定的迭代次数结束。

4.4 算法分析

上述的虚拟机放置问题把对物理机能源资源的优化抽象为装箱问题，把物理机抽象为箱子，虚拟机抽象为物品。对于大小不一的虚拟机映射到大小不同的物理机上，使得所需物理机的个数最少。另外，在虚拟机放置过程中，要考虑 CPU、内存、存储、接入带宽、输出带宽等各个类型资源的约束，因此，把此类问题抽象为多资源约束的装箱问题。众所周知，装箱问题是 NP 难问题[11]。它的时间复杂度为 $O(n^n)$，进一步地，考虑网络拓扑和当前的网络流量，进行网络资源的优化，虚拟机放置问题可以抽象为二次分配问题，最小化网络通信流量，使得流量通过网络设备的数量减少优化网络资源能源。二次分配问题也是 NP 难问题[12]。它的时间复杂度为 $O(n^n)$。把数据中心能耗优化问题抽象为装箱问题和二次分配问题的结合。而对网络性能的改进，利用最小化最大链路利用率实现，从流量工程的角度来看，这也是 NP 难问题。这个问题是经典的多目标组合优化问题，而求解此类问题的难点是如何降低时间复杂度和如何使得结果更接近最优解。本文设计两阶段贪婪算法求解此问题，较好地解决了这两者之间的平衡。

基于最小割的层次聚类算法的时间复杂度为 $O(n^3)$，基于 BF 的虚拟机放置算法的时间复杂度为 $O(n^2)$，解决虚拟机放置算法的总体时间复杂度为 $O(n^3)$。

5 实验仿真及结果分析

本文用 C++开发了 BF-HC 算法、BF-HC-LS 算法的仿真程序，解决装箱问题的最常见近似算法有 NFD(next fit decreasing)算法、FFD(first fit decreasing)算法、BFD 算法等[19]。因为 BF-HC 和 BF-HC-LS 算法是按照流量聚集之后，采用最佳适应（BF）的放置方法，所以选取与同样采用按虚拟机大小排序 BF 放置的 BFD 算法进行比较。另外，再选取流量分布较为平均的随机算法进行比较。

数据中心采用层次化拓扑结构，如多根树[7]、VL2[20]、Fat-tree[21]等，本文选用当前数据中心中最常见的树型拓扑，以及将来数据中心可能用到的 Fat-tree 拓扑。

仿真的基本输入主要包括三部分：虚拟机资源向量组、物理机资源向量组、虚拟机之间的流量矩阵。对于虚拟机资源向量组，Amazon EC2[22]提供了灵活的选择不同虚拟机的实例大小来满足不同的应用需求，参考选取 Amazon EC2 所提出的虚拟机大小和配置。对于虚拟机流量矩阵，本文的实验参照文献[10,23]的流量模式。通过它们的测量和估算，流量在较长的间隔期间内是相对稳定的。

能耗的计算方法基于式(1)。由于虚拟机整合，激活的物理机大多利用率较高，运行在高负载的情况下。为了便于计算和实验，把物理机能耗参数简化为高负载的功耗。网络设备的功耗也采用同样的处理方式。对于物理机和网络设备的具体功耗参数，可以根据设备的功耗说明来确定。在本实验中，物理机的功耗为 750 W。每个交换机的功耗有 80 W。网络带宽的容量为 1 000 M，交换机为三层交换机。

5.1 无网络拥塞情况

如果网络中无拥塞发生，本文的方案主要考虑以优化能耗为主，运行 BF-HC 算法减少数据中心的能耗，没必要运行 BF-HC-LS 算法。仿真实验根据数据中心作业所对应虚拟机数目的规模不同，选取了 100、200、300 个虚拟机各为一组实例，它们有不同的 CPU 大小、内存容量、磁盘空间、带宽等，在给定的一批物理机资源大小相同的情况下，对这 3 批虚拟机需求分别在 Tree 拓扑、Fat-tree 拓扑下用各类算法计算数据中心能耗和网络最大链路利用率。

在 Fat-tree 拓扑下，图 6 显示了 BF-HC 算法、BFD 算法以及随机算法对数据中心能耗的比较结果，可以看出不同的虚拟机与物理机的映射关系对数据中心的能耗影响是不一样的。由于随机算法所需的激活物理机数目和网络设备数目都比较多，所以数据中心能源消耗是最大的。BF-HC 算法所消耗的能源是最少的，BFD 算法次之。这里，BF-HC 算法与 BFD 算法所需的物理机个数是差不多的。BF-HC 算法的能耗之所以优于 BFD 算法，在于 BF-HC 算法对于网络总流量的优化使得其所需的激活网络设备的数目较少。图 7 列出了在 Fat-tree 拓扑下，这 3 种算法对数据中心网络总流量优化的结果，BF-HC 算法对于网络总流量的优化相比较

BFD 算法和随机算法有明显的优势。BF-HC 算法平均比 BFD 算法减少网络流量 29%，比随机算法减少 41%。而在 Tree 拓扑下，BF-HC 算法平均比 BFD 算法减少网络流量 65% ，比随机算法减少 81%。BF-HC 算法效果更明显，由于篇幅所限，这里没有列出 Tree 拓扑的仿真结果。

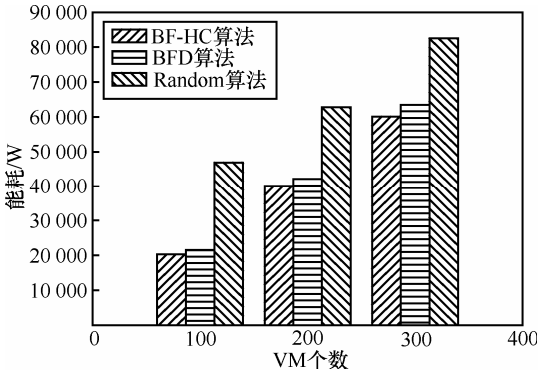


图 6　无拥塞情况下能耗比较

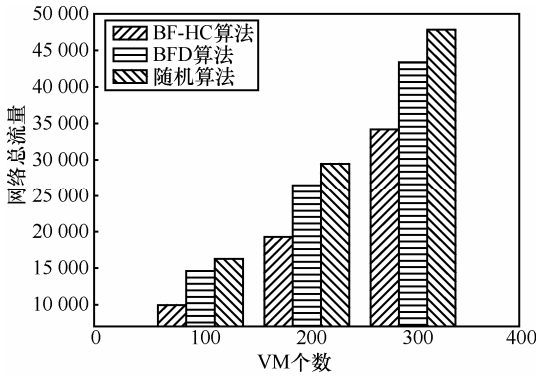


图 7　无拥塞情况下网络通信总流量比较

图 8 显示了在 Fat-tree 拓扑下，这 3 类算法对网络最大链路利用率的影响。可以看出随机算法由于流量分布较为平均，最大链路利用率较低。而 BF-HC 算法比 BFD 算法的最大链路利用率低，平均下降了 12%左右。

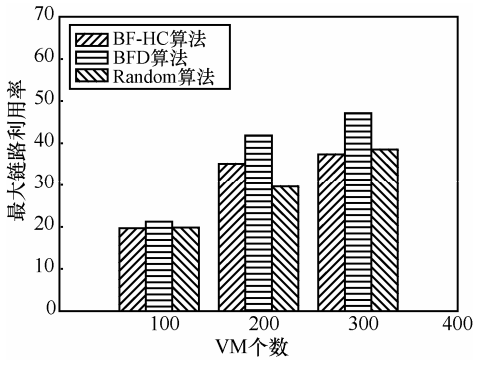


图 8　无拥塞情况下最大链路利用率比较

在本节的仿真实验中，相比较 BFD 算法，BF-HC 算法在优化网络总流量和最大链路利用率上有明显的优势，而且优化了数据中心的能源消耗。

5.2　有网络拥塞情况

如果有拥塞发生，笔者主要考虑以优化网络性能为主，在保证网络性能的前提下，最小化数据中心能耗。采用 BF-HC-LS 算法进行优化。选用热点链路数（HLN, hotspot link number）作为网络性能的衡量参数。

图 9~图 11 分别显示了在 Fat-tree 拓扑下，BF-HC-LS 算法、BF-HC 算法、BFD 算法和随机算法对能耗、网络总流量、HLN 数目的比较结果。5.1 节已经比较了 BF-HC 算法、BFD 算法和随机算法对能耗、总流量的影响，本节主要比较 BF-HC-LS 算法和 BF-HC 算法。在图 9 中，对于能耗的优化，BF-HC-LS 算法比 BF-HC 算法要差一些，平均增加了 6%，但差距不是很明显。同样，在图 10 中，对于总流量的优化，BF-HC-LS 算法比 BF-HC 算法也要差一些，平均增加了 8%。这是由于 BF-HC-LS 算法为了优化网络链路利用率，使得虚拟机之间的流量分布更为平均，使用了更多的物理机和交换机。

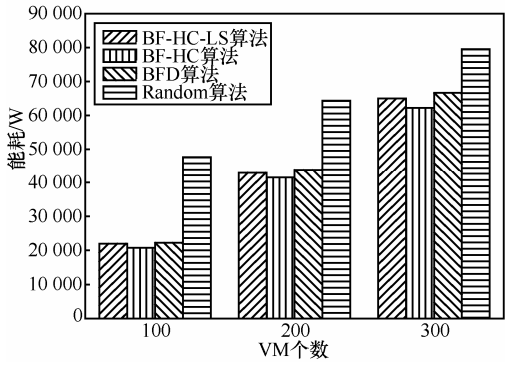


图 9　有拥塞情况下能耗比较

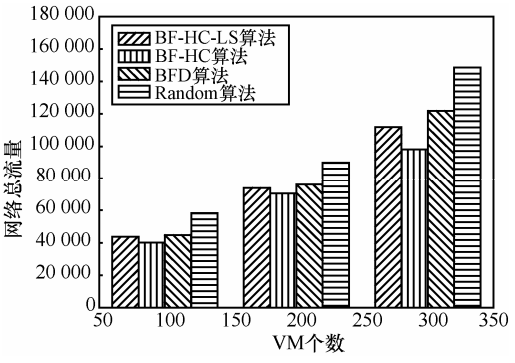


图 10　有拥塞情况下网络通信总流量比较

然而，在图 11 中，在优化 HLN 方面，BF-HC-LS 算法明显具有优势，例如，在 200 个虚拟机的情况下，相比较 BF-HC 算法，BF-HC-LS 算法的 HLN 减少了 8 条。虽然 BF-HC-LS 算法不能完全避免拥塞，但有效地减少了拥塞数。

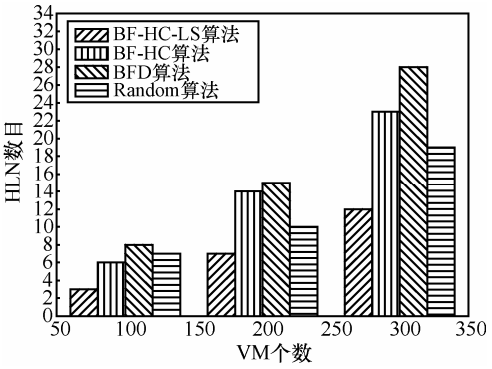


图 11 有拥塞情况下热点链路数比较

从这个仿真可以看出，相比较 BF-HC 算法、BFD 算法、随机算法，BF-HC-LS 算法在能耗没有增加太多的前提下，优化了网络性能，尽量避免了网络拥塞，在虚拟机放置过程中，较好地实现了能耗与网络性能的平衡。

6 结束语

虚拟机放置问题是目前云数据中心虚拟化技术研究的热点。本文提出了一种虚拟机在物理机上的放置方法，在物理机大小和网络链路容量的约束下，优化物理机与网络设备的资源利用率和能耗，并尽可能使得网络流量分布均衡，减少网络拥塞。提出了一种新的二阶段启发式算法。首先，如果没有产生拥塞，算法以优化能耗为主，利用最小割的层次聚类算法与 BFD 算法相结合来优化物理机和网络设备的能耗。第二，如果发生网络拥塞，以优化网络性能为主。利用局部搜索算法来最小化最大链路利用率，减少拥塞链路数目。仿真实验表明，相比较 BFD 算法、随机算法，在能耗变化不大的情况下，所提算法优化了网络流量分布，降低了网络的拥塞数目。

本文主要研究的是利用虚拟机放置技术，使得云计算数据中心网络性能与能耗达到平衡。而云计算数据中心的虚拟机放置完毕后，随着负载的变化，虚拟机大小以及虚拟机与物理机的映射关系也随之发生变化，这就要考虑虚拟机迁移的问题。在不影响业务性能的前提下，如何最小化虚拟机迁移代价实现虚拟机的动态调整是下一步的研究工作。

参考文献：

[1] BARHAM P, DRAGOVIC B, FRASER K, *et al*. Xen and the art of virtualization[J]. ACM SIGOPS Operating Systems Review, 2003, 37(5): 164-177.

[2] Amazon EC2[EB/OL]. http://aws.amazon.com/ec2.

[3] VERMA A, AHUJA P, NEOGI A. PMapper: Power and Migration Cost Aware Application Placement in Virtualized Systems[M]. Springer, 2008.

[4] BOBROFF N, KOCHUT A, BEATY K. Dynamic placement of virtual machines for managing sla violations[A]. Proc Integrated Network Management, IEEE[C]. Munish,Germany, 2007.119-128.

[5] CARDOSA M, KORUPOLU M R, SINGH A. Shares and utilities based power consolidation in virtualized server environments[A]. Proc Integrated Network Management,IEEE[C]. New York, USA, 2009. 327-334.

[6] WANG M, MENG X, ZHANG L. Consolidating virtual machines with dynamic bandwidth demand in data centers[A]. INFOCOM 2011, IEEE[C]. Shanghai, China,2011.71-75.

[7] AL-FARES M, LOUKISSAS A, VAHDAT A. A scalable, commodity data center network architecture[J]. ACM SIGCOMM Computer Communication Review, 2008,38(4):63-74.

[8] MANN V, KUMAR A, DUTTA P, *et al*. VMFlow: Leveraging VM Mobility to Reduce Network Power Costs in Data Centers[M]. Springer: Berlin Heidelberg, 2011. 198-211.

[9] FANG W, LIANG X, LI S, *et al*. VMPlanner: optimizing virtual machine placement and traffic flow routing to reduce network power costs in cloud data centers[J]. Computer Networks, 2013,57(1):179-196.

[10] MENG X, PAPPAS V, ZHANG L. Improving the scalability of data center networks with traffic-aware virtual machine placement[A]. INFOCOM 2010,IEEE[C]. San Diego,CA,USA,2010. 1-9.

[11] MAN JR E C, GAREY M R, JOHNSON D S. Approximation Algorithms for Bin Packing: A Survey[M]. Approximation Algorithms for NP-Hard Problems, 1996.46-93.

[12] WOEGINGER G J. Exact Algorithms for NP-Hard Problems: A Survey[M]. Springer: Combinatorial Optimization-Eureka, 2003.

[13] DEB K. Multi-Objective Optimization[M]. Springer: Multi-Objective Optimization Using Evolutionary Algorithms, 2001.13-46.

[14] JIANG J W, LAN T, HA S, *et al*. Joint VM placement and routing for data center traffic engineering[A]. INFOCOM 2012, IEEE[C].Orlando, Florida,USA, 2012. 2876-2880.

[15] GUPTA A, KALÉ L V, MILOJICIC D, *et al*. HPC-aware VM placement in infrastructure clouds[A]. IEEE Intl Conf on Cloud Engineer-

ing[C]. San Francisco,California,USA, 2013. 11-20.

[16] BELOGLAZOV A, BUYYA R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers[J]. Concurrency and Computation: Practice and Experience, 2012, 24(13): 1397-1420.

[17] BIENKOWSKI M, FELDMANN A, JURCA D, *et al*. Competitive analysis for service migration in vnets[A]. Proceedings of the Second ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures[C]. New Delhi, India, 2010.17-24.

[18] HELLER B, SEETHARAMAN S, MAHADEVAN P, *et al*. ElasticTree: saving energy in data center networks[A]. Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation[C]. San Jose, California, USA, 2010.19-21.

[19] JOHNSON D S, DEMERS A, ULLMAN J D, *et al*. Worst-case performance bounds for simple one-dimensional packing algorithms[J]. SIAM Journal on Computing, 1974, 3(4): 299-325.

[20] GREENBERG A, HAMILTON J R, JAIN N, *et al*. VL2: a scalable and flexible data center network[J]. ACM SIGCOMM Computer Communication Review, 2009,39(4):51-62.

[21] NIRANJAN MYSORE R, PAMBORIS A, FARRINGTON N, *et al.* PortLand: a scalable fault-tolerant layer 2 data center network fabric[J]. ACM SIGCOMM Computer Communication Review, 2009,39(4):39-50.

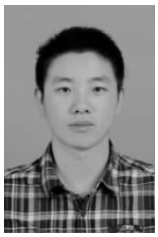
[22] Amazon EC2[EB/OL]. http://aws.amazon.com/ec2/instance-types/.

[23] BENSON T, AKELLA A, MALTZ D A. Network traffic characteristics of data centers in the wild[A]. Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement[C]. Melbourne, Australia,2010.267-280.

作者简介：

董健康（1973-），男，甘肃陇南人，北京邮电大学副教授，主要研究方向为云计算、数据中心网络。

王洪波（1975-），男，河北唐县人，博士，北京邮电大学副教授，主要研究方向为云计算及数据中心网络、互联网测量与管理、下一代互联网体系结构及新应用。



李阳阳（1987-），男，江苏扬州人，北京邮电大学博士生，主要研究方向为数据中心网络资源管理及调度。

程时端（1940-），女，上海人，北京邮电大学教授、博士生导师，主要研究方向为下一代互联网技术、互联网 QoS 技术、数据中心网络。