Virtual Machine Scheduling for Improving Energy Efficiency in IaaS Cloud

DONG Jiankang, **WANG Hongbo**, **LI Yangyang**, **CHENG Shiduan**

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China

Abstract: In IaaS Cloud, different mapping relationships between virtual machines (VMs) and physical machines (PMs) cause different resource utilization, so how to place VMs on PMs to reduce energy consumption is becoming one of the major concerns for cloud providers. The existing VM scheduling schemes propose optimize PMs or network resources utilization, but few of them attempt to improve the energy efficiency of these two kinds of resources simultaneously. This paper proposes a VM scheduling scheme meeting multiple resource constraints, such as the physical server size (CPU, memory, storage, bandwidth, etc.) and network link capacity to reduce both the numbers of active PMs and network elements so as to finally reduce energy consumption. Since VM scheduling problem is abstracted as a combination of bin packing problem and quadratic assignment problem, which is also known as a classic combinatorial optimization and NP-hard problem. Accordingly, we design a two-stage heuristic algorithm to solve the issue, and the simulations show that our solution outperforms the existing PM- or network-only optimization solutions.

Key words: IaaS cloud; virtual machine scheduling; energy efficiency; bin packing problem; quadratic assignment problem

I. INTRODUCTION

Cloud computing provides users with on-demand, flexible, reliable and low-cost services, and the infrastructure of these services is cloud data centers [1]. Cloud providers need to construct and manage data centers with low cost. With the increasing scale of cloud computing, energy consumption is undoubtedly growing, which increases operation cost. The related report from Microsoft [2] shows that the physical resources in data center (e.g. CPU, memory, storage, etc.) will account for 45% of the total cost, and energy consumption will account for 15%; according to [3], in the past five years, energy consumption in data centers has been doubled. Therefore, how to reduce energy consumption is becoming an important issue in data centers.

Lower server utilization causes a huge waste of electricity, The data collected from more than 5000 production servers over a six-month period have shown that although servers usually are not idle, the utilization rarely approaches 100% [4]. Most of the time servers operate at 10%-50% of their full capacity, leading to extra expenses on over-provisioning and extra Total Cost of Acquisition (TCA) [4]. Moreover, energy consumption brings high cost in ancillary cooling facilities, which will undoubtedly increase operation cost, so it is an urgent requirement to find out new solutions to reduce energy consumption in data centers.

Now, most of physical servers in cloud data center use virtualization technology, which runs multiple virtual servers on the same physical machine (PM) in order to improve resource utilization and reduce energy consumption. Besides, virtualization also helps cloud providers to achieve flexible and effective management.

For public cloud with virtualization, one of its major services is infrastructure as a service (IaaS), such as Amazon EC2 [5]. Tenants pay to rent virtual machine (VM) resources, based on service level agreements (SLAs), cloud providers take advantage of VM's flexible placement on PM to optimize the resources allocation so as to meet the demands of tenants. Since different resource utilization is caused by different mappings between VMs and PMs, so for cloud providers, the main issue should be how to place multiple VMs demanded by tenants onto physical servers so as to minimize the number of active physical resources and reduce energy consumption, and correspondingly, operation and management costs will be reduced. Nowadays, VM scheduling is becoming a hot issue.

Currently, the first challenge for VM scheduling problem is how to optimize the energy consumption of both PMs and network elements (switches, routers, links, etc.). There are mainly two research types according to different resources optimization: one is to consolidate VMs to improve PM utilization [6-9], but these studies do not consider the impact of network topology and current communication traffic. Actually as the scarce resources in data center, network resources have a direct impact on application performance [10]. The other type uses VM scheduling to optimize network traffic [11, 12], but these studies simply assume that sufficient resources are provided when placing VM to PM, and neglect the issue on how to optimize PM's CPU, memory, storage etc.. In fact, two factors should be considered at the same time when it comes to VM scheduling: the allocation of PM resources and the allocation of network resources. VM scheduling optimizing both PM and network resource utilization can effectively reduce energy consumption without impacting on application performance.

The second challenge is how to design a two-stage VM scheduling in static placement and dynamic migration [7]. Static placement refers to cloud providers consider to place VM on idle PM to satisfy the resource demands based on VM vector, and static placement is generally applied for initial VM placement. Dynamic migration refers to VMs demand to be readjusted when VMs do not correspond with PMs in the initial calculation, especially when VMs meet workload fluctuations in running so that VMs will dynamically migrate [13]. Static placement concerns the accuracy of the objective function, while dynamic migration mainly concerns how to minimize the migration cost, so different VM schedulings should be taken in these two stages.

This paper proposes an optimal VM scheduling scheme on the basis of multiple resources constraints by cross-optimizing VMs placed on PMs to minimize the numbers of activated PMs and network elements, thereby reducing the energy costs in data center. Based on this, we propose a two-stage scheduling strategy: (1) In static VM placement, the optimization of the PM resources is abstracted as a multi-dimensional bin packing problem (BPP) [14] to reduce the number of activated PMs; while for the optimization of network resources, the network topology and the current network traffic is abstracted as the quadratic assignment problem (QAP) [15] so as to put the large traffic between VMs into the same PM or the same network switch. Once network communication cost is small, the number of required network elements will be reduced [16]. As is well known, BPP and QAP is NP- hard problem [14, 15], so we take a new greedy algorithm to solve. (2) In dynamic VM migration, the number of migrated VMs is taken as migration costs. In a given number of VM migration, we attempt to optimize the network performance and energy consumption, and apply a new heuristic algorithm to solve.

The main contributions of our paper are: (1) We present a multi-resource constraint VM scheduling scheme to improve energy efficiency of physical servers and network elements. (2) We design a two-stage heuristic algorithm for VM scheduling in static placement and dynamic migration. (3) Compared with the schemes only optimizing PM or network resources, our algorithm has achieved better results in the simulation.

Our paper is organized as follows: Section II presents the related work. VM scheduling is described and modeled in Section III. Section IV puts forward VM placement and migration algorithm. The simulation is shown in Section V. Section VI concludes the paper.

II. Related Work

There are two focuses on VM scheduling problem. One is to consider how to place VM in accordance with the physical servers [6-9]. Verma et al. [6] dynamically re-adjust server's location and consider the cost of application migration and energy with a simple algorithm, it shows that dynamic migration technology realizes low energy cost. Bobroff et al. [7] adopt prediction techniques while minimizing the number of active PMs, and present mechanism for dynamic migration of VMs based on a workload forecast. Dong et al. [8] propose a static VM placement scheme to reduce both the numbers of PMs and network elements in a data center to reduce energy consumption, but this scheme does not consider dynamic VM migration. Wang et al. [9] consider the consolidation of VM bandwidth with PM bandwidth as a random packing NP-hard problem (SBP), it shows certain size of VM is loaded onto a PM with a probability distribution, and the goal of optimization is minimizing the number of PMs. However, [6-7,9] only consider PM optimization, and ignore the optimization of other resources.

The PM optimizing schemes above either consider CPU constraints [6] or PM bandwidth constraints [9], and they neglect network topology and VM network traffic. The other type considers how to place VM to optimize network resources [11, 12]. Meng et al. [11] propose to improve the network scalability in data center network with a traffic-aware VM placement scheme. By optimizing VM's location in the host, the traffic between VMs is related to the network physical distance, and VMs with large traffic can be placed on PMs nearby to reduce the total network traffic. Mann et al. [12] propose to reduce energy consumption by VM migration technology and network routing optimization. Such solutions only assume to meet needs of physical servers, and they only optimize network resources and neglect the optimization of physical server resource.

Currently, some studies consider how to place VM with the multi-resource constraints [17-19]. Singh et al. [17] take advantage of VM migration technology to change VM's position in PM so as to achieve load balance in system performance, and such problem is abstracted as multi-dimensional knapsack problem. However, this approach is different from our proposal, our proposal attempts to reduce energy consumption, so it is not suitable to apply the approach in [17]. Chaisiri et al. [18] propose an optimal VM placement algorithm. This algorithm can minimize the cost spending in each plan for hosting VMs in a multiple cloud provider environment under future demand and price uncertainty. It is also different from our optimization goal. Ferreto et al. [19] propose a linear program to optimize VM's location, and design a heuristic algorithm to control VM migration; Liao et al. [20] propose a dynamic VM mapping in the cluster system and data center to optimize energy, and this dynamic VM mapping can realize VM migration without much influence on application performance. [19, 20] use VM migration to realize dynamic server consolidation, and such proposals are more suitable for cluster system or small VMs than for large VMs in the data center, while our solution is applicable for static VM placement with large VM numbers.

III. Design and Model

3.1 Problem description and symbol definition

In IaaS, cloud providers lease resources for tenants, and tenants sign SLAs with cloud providers to guarantee the service performance. Our VM scheduling scheme is mainly for cloud providers without violating SLAs as possible, that is, on the premise of ensuring the performance. The main concern for cloud providers should be how to design VM scheduling scheme to improve the physical resource utilization in the resource pool and reduce the numbers of active PMs and network elements, so our scheme will finally reduce the operation cost in data center by means of reducing the hardware investment and energy consumption.

Assume each tenant requires N VMs. VM i resource demand is a d-dimensional vector $\vec{S}_i$, and each dimension represents certain type of VM resources (such as CPU, memory, disk etc.). For vector $\{S_{i,1}, S_{i,2}, ..., S_{i,d}\}$, d is the number of types of resources. For example, $S_{i,2}$ represents the desired value of the resource type 2 in VM i. Vector set $\{\vec{S}_i\}_{i=1,...N}$ represents all VM resource demands, The main meanings of the symbols are given in Table I.

Similarly, PM set can be expressed as $P = \{P_1, P_2, ..., P_M\}$, PM $m_{=1,...,M}$ is also a d-dimensional vector, $\vec{H}_m$ represents the corresponding value of $\{H_{m,1}, H_{m,2}, ..., H_{m,d}\}$. P_m is the VM set on PM m.

3.2 Problem formalization

3.2.1 Optimization of server resources

Our focus is mainly on how to minimize number of PMs when different sizes of VMs are mapped to PMs with different sizes. Meanwhile, some resource constraints such as CPU, memory and storage should be considered. Based on this, we consider our problem as a multi-resource constraint bin packing problem, and our objective is to minimize the number of active physical servers. Here, we use the number of PMs to describe server energy cost. The fewer PMs will bring less energy consumption

We define $X_{i,m}$ as a binary variable, expressed as

$$X_{i,m} = \begin{cases} 1 & if \quad VM\ i\ on\ PM\ m \\ 0 & otherwise \end{cases} \tag{1}$$

$$\mathbb{F} = \{X_{i,m} | X_{i,m} \in \{0,1\}, \sum_{m=1}^{M} X_{i,m} = 1, \forall i\}$$

Set $\mathbb{F}$ represents a set of all VM choice, $\sum_{m=1}^{M} X_{i,m} = 1$ means each VM can only be placed on a single PM. Our goal is to minimize the number of PMs with an effective VM placement, and it can be formalized as follows:

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

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

$$\{X_{i,m}\} \in \mathbb{F}$$

Binary variables $Y_m \in \{0,1\}$ shows PM m is running or to be activated. The constraint is the number of multiple VMs placed on a PM cannot exceed the number of corresponding PM resources. E_{ser}^m is the energy consumption of the PM m.

E_{ser}^m is closely related to the numbers and the

Table I *Key notation and its meaning*

Symbol	Description
M	Number of PMs, indexed by $m = 1, ..., M$
N	Number of VMs, indexed by $i = 1, ..., N$
$\vec{\boldsymbol{H}}_m$	d dimensional resource vector of PM m ,its value $\{H_{m,1}, H_{m,2}, ..., H_{m,d}\}$, d is the number of resource types
$\vec{\boldsymbol{S}}_i$	D-dimensional resource vector of VM i its value $\{S_{i,1}, S_{i,2}, ..., S_{i,d}\}$
Y_m	Binary variable , 1 indicates PM m is in the activation status ; 0 indicates that PM m is sleep
$X_{i,m}$	Binary variable , 1 indicates VM i is placed on the PM m, whereas 0
P_m	VM set are placed on the PM m
E_{ser}^m	Energy consumption of the PM m.
$\boldsymbol{A}$	Communication traffic matrix, $(a_{i,j})_{N \times N}$ is the traffic between VM i and VM j
$\boldsymbol{B}$	Communication cost matrix, The communication cost between the PMs
$\pi(i)$	Mapping function , PM on which VM i is placed

loads of VMs, which directly affects the loads of PMs. The larger the number of VMs, the larger the loads of PMs. E_{ser}^{m} is modeled in Formula (3) and (4). In the Formula (3), $P(u)$ represents the power consumption of PM, and the general value of k is 0.7, which means that the power consumption of the idle PMs accounts for 70% of the maximum power consumption $P_{\max}$. Formula (3) shows that the idle PMs still cause large power consumption. u represents the PM utilization. As DVFS (Dynamic Voltage and Frequency Scaling) is only available to CPU, u mainly refers to CPU's utilization. u and $P(u)$ are in a linear relation.

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

$$E_{\text{ser}}^{m} = \int_{t} P(u(t))dt \tag{4}$$

3.2.2 Optimization of network resources

For the optimization of network resources, our objective is to minimize traffic in data center network, and we abstract this problem as QAP. We converge the large traffic between VMs onto same PM or on same switch. If the total communication traffic in network is smaller, then the number of network elements will be reduced, and the other idle network elements will be in a sleep state, finally the energy consumption will be reduced. Here, we use network communication traffic to describe network energy consumption. The proposal not only can save energy of the network elements, but also save the network bandwidth.

Traffic matrix $A = (a_{i,j})_{N \times N}$, communication cost matrix $B = (b_{h,p})_{M \times M}$, $a_{i,j}$ is the traffic between VM i and VM j, $b_{h,p}$ represents the communication cost between PM h and PM p, communication cost means the switch number the traffic passes between PMs. The larger the switch number, the greater the communication cost, the higher the network energy consumption.

Our goal is to find a mapping function $\pi(i)$ to meet VM i placed on PM. A VM can be placed on a PM, but a PM can place multiple VMs.

The objective function can be formally expressed as:

$$\text{Min } Cost_{net} = \sum_{i,j=1}^{N} a_{i,j} b_{\pi(i)\pi(j)} + \sum_{i=1}^{N} e_i g_{\pi(i)} \tag{5}$$

e_i is the traffic between VM i and external communications. $g_{\pi(i)}$ represents the communication cost between PM on which VM i is placed and the external switch . The function of the second portion is assumed to be a fixed number, due to the special nature of the network topology, it can be ignored in the algorithm.

3.2.3 Migration cost

There are many migration cost metrics [21], and we use relatively simple and effective one. We use the number of VMs to represent migration costs, the less the number of migrated VMs, the less the migration costs, the less the impact on the application. N_{mig} represents the number of VMs to be migrated.

$$\text{Min } Cost_{mig} = N_{mig} \tag{6}$$

Our goal is to minimize migration cost, MLU and total energy consumption, which is a multi-objective optimization problem [22].

$$\text{Min } f = Cost_{ser} + \alpha \cdot Cost_{net} + \beta \cdot Cost_{mig} \tag{7}$$

We normalize the values of $Cost_{ser}$ and $Cost_{net}$, which determine the energy consumption. The smaller the value of function f, the lower the energy consumption in data center. This is a multi-objective optimization problem, and it is also a classic combinatorial optimization problem.

IV. ALGORITHM

In Formula (7), $Cost_{ser}$ and $Cost_{net}$ are considered for static placement; $Cost_{ser}$, $Cost_{net}$ and $Cost_{mig}$ are considered for dynamic migration. Based on the static placement and dynamic migration, we design a two-stage heuristic algorithm: firstly, in VM placement, we propose to meet the physical capacity and network bandwidth capacity in order to minimize the energy consumption in the data center; secondly, in VM migration, we propose to minimize the migration costs to optimize the network maximum link utilization (MLU) and reduce the energy consumption.

4.1 VM placement algorithm

In this algorithm design, Formula (7) considers the energy consumption of servers $Cost_{ser}$ and network elements $Cost_{net}$ regardless of the migration costs $Cost_{mig}$. We attempt to achieve two main purposes: (1) The optimization of the energy consumption of PMs mainly by minimizing the number of activated PMs according to VM demand vector groups. (2) The optimization of the energy consumption of network elements by putting VMs with large traffic onto the nearby PMs to optimize the distribution of network traffic, so as to minimize the number of activated switches. As mentioned earlier, these two problems are abstracted as QAP and BPP, which is not only a multi-objective optimization problem but also a NP-hard issue. Generally, heuristic intelligent algorithms such as genetic algorithm, ant colony algorithm etc. are applied to solve this problem, but these algorithms have the defects such as poor time performance and the instable results, so a new greedy algorithm is applied in our paper for solving this multi-objective optimization problem.

The basic inputs are: tp: network topology, traffic routing, $\boldsymbol{A}$: traffic demands between VMs, $\vec{S}_i$: VM i demands vector, $\vec{H}_m$: PM m vector. The basic outputs are: X: the mappings of VMs on PMs, pm: the number of required PMs, sw: the number of required switches.

Our design is as followings: (1) With the flow between VMs in descending order, we propose to put the VM pairs with the large flow on the same PM. (2) Our scheme looks for one of VMs (in the rest of VMs) with largest communication flow which has been placed on PMs, and then places this VM on same PM. (3) If the current capacity of PM cannot meet the demands of the VM, activate another PM which keeps the shortest distance with the current PM. Meanwhile, according to the routing algorithm, our scheme looks for one of VMs (in the rest of VMs) with largest communication flow which has been placed on PMs, and then places this VM on same PM. (4) If the flows of VM to be placed cannot meet the network link bandwidth capacity, repeat the second step and re-activate a new PM.

In the hierarchical topology, routing algorithm mostly takes random selection strategy which evenly distributes flows. However, we take greedy knapsack algorithm to evaluate possible paths and choose the leftmost route with sufficient free capacity in a certain hierarchy layer of a structured topology, such as a fat-tree. Within a layer, the paths are chosen in a deterministic left-to-right order other than a random order. When all flows are allocated, the algorithm returns to an active network elements subset where the traffic goes through, and the network elements without flows can be in the sleeping or closed state. Therefore, we combine VM placement with flow path routing to the energy-saving objective. This algorithm is defined as VM-P. The algorithm is described in Algorithms 1.

Algorithms 1 VM-P algorithm

Input: tp, A, B, $\vec{S}_i$,$\vec{H}_m$
Output : X, pm, sw
Initial X,A,B,tp
Select the current PM
pm=1
The flows of VM pairs in descending order according to A
Pick VM s and VM t pairs with the largest flows
Place VM s and VM t on PM m
While VM ∈ VMset cannot be placed
Calculate traffic between VMs on PM m and VM s not yet placed
Choose VM s with the largest communication flows
If The capacity of VM s less than capacity of PM m
Calculate and Activate PM new
  pm++
  PM m ← PM new
 else
 If VM s cannot meets the capacity of network bandwidth
  Calculate and Activate PM new
  pm++
  PM m ← PM new
 Endif
Endif
Place VM s onto PM m
Endwhile
Calculate sw according to A,tp,Routing algorithm
Output X,pm,sw

4.2 VM migration algorithm

With the time changing, VM workloads and the traffic between VMs may also change. Thus, in order to adapt to such change and the flexibility, to meet the needs of the tenants and to maximize the interests of the cloud provider, we propose an algorithm for VM migration. According to Formula (7), not only are the energy consumption of PM $Cost_{ser}$ and network elements $Cost_{net}$ considered, but also the migration costs $Cost_{mig}$ is considered.

The current position of VMs on the PMs is taken for mapping matrix X, matrix A represents the current traffic with the changed VM, tp expresses the network topology, parameter tv expresses the migration costs threshold (the number of VM migration); the return value is formed with the optimized VM target mapping matrix X` and the migration costs N_{mig}.

The main points of our algorithm are: (1) we design VM-P algorithms to meet X^{target}, which refers to capacity limitations of PMs and the network bandwidth, and then calculate set C which meets the optimal demands and needs to be migrated. If the number of VMs to be migrated is less than tv (the threshold value), then migrate according to X^{target}, and the migration task ends. (2) if the number of VMs to be migrated is more than tv, then set up a number of iterations *nMax*, and conduct a number of iterations to select the better placement X`. (3) in each iteration, randomly select a VM, and migrate to the corresponding PM in accordance with the mappings of X^{target}; and then delete this VM from the current PM, and define this PM as PM_{mig} (4) X^{target} finds out the VM mapped to PM_{mig}, and the mapping of this VM in X is PM_{ori}; migrate VM to PM_{mig}, repeat the iteration from PM_{ori} to PM_{mig}, until the migration times reach tv. This algorithm is defined as VM-Mig. The algorithm is described in Algorithm 2.

4.3 Algorithm analysis

4.3.1 VM-P algorithm

We abstract the energy optimization of PMs as BPP, PMs as boxes and VMs as items. When VMs with different sizes map onto PMs with different sizes, the least number of PMs is required. In addition, various resources constraints such as CPU, memory, storage, bandwidth have to be taken into account. Therefore, such problem is defined as a multi-resource constraints bin packing problem, which is also NP-hard problem [14]. Its time complexity is $O(n^n)$. Furthermore, network topology and communication traffic are also considered to optimize network resources, which can be abstracted as QAP, which is NP-hard problem [15].

Now, the common approximation algorithms for VM placement problem are Next Fit (NF), First Fit (FF), Next Fit Decreasing

Algorithms 2 VM-Mig algorithm

```
Input: X, tp, A, B,tv
Output : X`,N_mig
X^target=VM-P(A,tp)
C=Diff(X^target,X)
If C.size<=tv:
X`=X^target
  Return X`,N_mig
Else:
  X^tmp=X
P=get_performance(X,A,tp)
 For i <nMax
    N_mig=0
    tmp=i
    While(N_mig<tv):
      Move_vm(X^tmp,X^target,C[tmp])
      N_mig++
      Next=vm_on_target_pm(X^target,C[tmp])
      Erase(C,tmp)
      If next==NULL:
        tmp=0
      Else:
        tmp=next
      Endif
    Endwhile
    P`= get_performance(X^tmp,A, tp)
    If P` better_than P:
      P=P`
      X`=X^tmp
    Endif
 Endfor
Return  X`, N_mig
End if
```

(NFD), First Fit Decreasing (FFD), etc [20]. FFD is the most common algorithm, which sorts VM resource size according to the descending order, and firstly places VM with the largest size. A new VM will be put onto the firstly used PM, only when this PM has overloaded, next PM can be used. The time complexity of FFD is O(n), and the space complexity is O(1). Our algorithm both considers the network traffic and computing resource optimization. Compared with FFD, the time complexity of VM-P is rather higher, its time complexity is $O(m \cdot n^2)$, and space complexity is $O(n^2)$.

4.3.2 VM-Mig algorithm

VM-Mig algorithm uses local search principle. Local search is a common algorithm to solve NP-hard problem. Despite its defects such as the poor stability and possible local optimum, local search algorithm is widely used in engineering practice for its fast and better accuracy. The performance of VM-Mig mainly depends on the number of iterations *nMax*. The greater the value of *nMax*, the more accurate the result. When *nMax* reaches a certain value, the algorithm is convergent. The time complexity of VM-Mig is $O(nMax \cdot n^2)$, and the space complexity is $O(n^2)$.

V. Evaluation

5.1 Simulation setup

Hierarchical topologies are usually applied in data center, such as multi- root tree [10], VL2 [21], and fat-tree [23] etc., and we choose the most common one —fat-tree, which may also be commonly used in future data center.

We use C++ to develop our VM-P and VM-Mig algorithms. In VM placement patterns, FFD is commonly used to solve BPP, which can only optimize the energy consumption of PMs; T-opt is taken to solve QAP, which only optimize the energy consumption of the network resources; while Random algorithm can neither optimizes the energy consumption of PMs nor that of network resources. Based on this, we respectively compare our VM-P algorithm with FFD, T-opt and Random. And then in VM migration patterns, we compare VM-P with VM-Mig algorithm.

5.2 Results

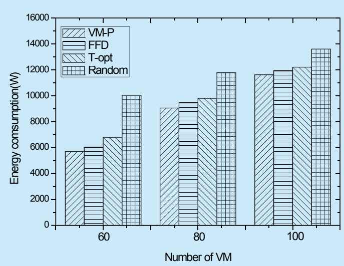
5.2.1 VM placement patterns

In the initial placement of VMs, our scheme mainly attempts to optimize energy consumption by operating VM-P algorithm. With the different scale of the VMs in the data center, a group of VMs 60, 80 and 100 are respectively selected in our simulation, and 16 PMs are also selected. The basic inputs in our simulation are: VM resource vector group, PM resources vector group and traffic matrix between the VMs. For VM resource vector group, Amazon EC2 [5] provides a flexible choice to meet different application needs, so we select the size of VMs and configuration according to Amazon EC2. For VM traffic matrix, our experiments take the traffic patterns in [11], and through the measurements and estimation in [24], they show that the traffic at long intervals is relatively stable. Two traffic models exist in the data center ---global model and partitioned model. Under the global traffic model, each VM sends traffic to every other VM at equal and constant rate, whereas this sending rate can differ among VMs. Under the partitioned traffic model, each VM belongs to a group of VMs and it sends traffic only to other VMs in the same group. Thus, we take the partitioned model for the pairwise traffic rate following a normal distribution.

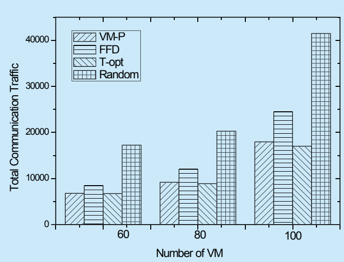
To simplify the simulation, we use a homogenous PMs and VMs, that is, the VMs in our simulation have the same CPU size, memory capacity, disk space, etc.. We apply Random, FFD, T-opt and VM-P to these three groups of VM to calculate the required number of PMs, switches, and then we calculate the energy consumption, total communication traffic and MLU differences in fat-tree topology by three algorithms.

Fig.1-(a) shows the energy consumption of these four algorithms in the fat-tree topology.

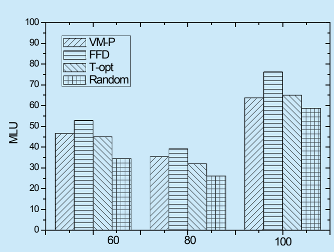
It can be seen that different mappings between VMs and PMs have different effects on energy consumption. The energy consumption is calculated on the basis of equation (7). The power consumption of each PM is 750watt, and the power consumption of each switch is 80watt. Since the random algorithm requires more number of activated PMs and network elements, the energy consumption by random algorithm is the greatest; the energy consumption by VM-P is the smallest, FFD and T-opt is just between these two. Here, VM-P and FFD require almost the same number of PMs, but VM-P shows better effect on the optimization of the total communication traffic than FFD, which leads to less number of network elements. More activated PM caused by T-opt leads to the larger energy consumption than by VM-P.



(a)Energy *Consumption*



(b) *Total Communication Traffic*



(c) *MLU*

Fig.1 *Algorithm comparison in VM placement patterns*

Fig.1-(b) shows the differences in total network traffic optimization by the three algorithms in the fat-tree topology. VM-P shows better effects compared with FFD and random algorithm. The amount of network traffic by VM-P is averagely decreased by 41% than by FFD, and decreased by 85% than by Random. Compared with T-opt, VM-P is weaker in optimizing the network traffic, but the gap between these two is quite slight.

Fig.1-(c) shows the MLU differences by the three algorithms in fat-tree topology. It can be seen that random algorithm has a more even traffic distribution, but MLU is quite low; MLU by VM-P is 10% lower than by FFD. There is almost no gap between T-opt and VM-P in optimizing MLU.

Compared with FFD, VM-P in our simulation shows greater advantage in optimizing MLU and total network traffic, and the objective of energy-saving is also achieved. Compared with T-opt, VM-P achieves better energy efficiency.

5.2.2 VM Migration Patterns

With the workloads and flows between VMs changing, we use VM-Mig algorithm to realize the dynamic migration of VMs. We selected 16 PMs and 60 VMs, and the other conditions are similar to our simulations above. Due to the limited writing space, the repetitive details will not be listed. As the changes of total network traffic, energy consumption will also be changed, so we adopt more intuitive total network traffic as evaluation index, and the optimization results of energy consumption will not be repeated.

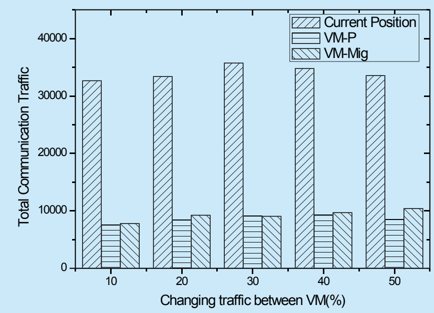
Fig.2-(a) shows the comparisons of total communication traffic by Current Position, VM-P and VM-Mig; and Fig.2-(b) is the comparison of MLU. X axis shows the changing percentage in the traffic between VMs; the migration costs mainly refer to the number of VMs to be migrated. Here, the migration threshold is about 20% of the total number of VMs. The total communication traffic and MLU are better optimized by VM-P, but a large number of VMs are not in the initial po-

sition, which means more VMs have to be migrated, and it is not practical in real operating. If we do not migrate and maintain the initial mapping, a poor optimization of energy and MLU will re-occur. Thus, VM-Mig is applied to solve this problem, under the condition of migrating 20% of total number of VMs, and the results are almost close to the effect caused by VM-P.

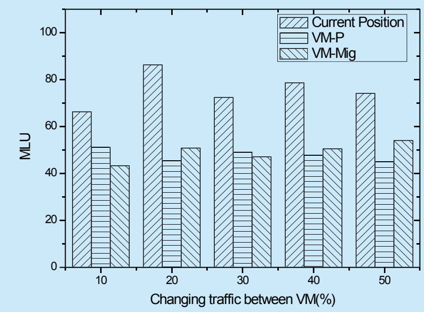
Under the condition of the traffic change, we assume 20% change to see the migration costs' influence on total communication traffic and MLU. Fig.3-(a) shows the comparisons in total communication traffic, Fig.3-(b) shows the comparison in MLU, and X-axis shows the changes of the migration cost. As can be seen from the figures, when the migration threshold is up to 10%, VM-Mig achieves better optimization results. Compared to the current position, the total network traffic by VM-Mig drops, which is almost close to the results by VM-P. However, VM-P easily causes a large number of VM migrations. With the increasing migration threshold, VM-Mig does not show a linear increase, and the change is rather slight. In addition, MLU by VM-Mig also drops. Compared with VM-P and current position, VM-Mig can effectively optimize the network performance, which preferably realizes the balance between energy consumption and network performance with a smaller migration cost.

VI. Conclusions

In our VM scheduling scheme, we consider multi-resource constraints of PM and network bandwidth and attempt to save energy consumption. We propose to optimize both PM resources energy and network resources energy. We abstract VM placement problem as a combination of multi-constraint BPP and QAP. After analyzing the advantages and disadvantages of the selected algorithm, we propose a new two-stage heuristic algorithm to solve such problem. Our simulation results show that the algorithm has achieved better

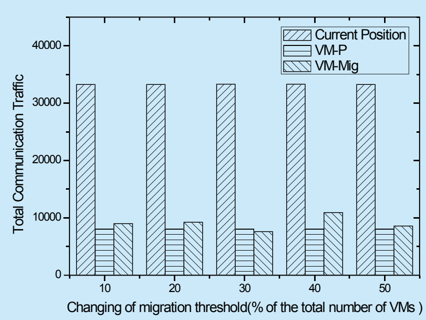


(a) *Total Communication Traffic*

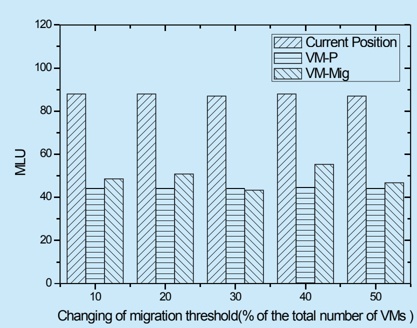


(b) *MLU*

Fig.2 *Changing traffic in VM migration patterns*



(a) *Total communication traffic*



(b) *MLU*

Fig.3 *Changing migration costs in VM migration patterns* results.

The main aim in our paper is to reduce the quantity of physical resources to save energy consumption in data center, but there are still some potential problems. On the one hand, if more and more VMs are placed on same PM, physical resources will overload, which may have influence on VM resource expansion. On the other hand, if more network traffic aggregates on the same network link, network link utilization will be improved, but it will also bring network congestion. Thus, a balanced workload for PMs should be a major future concern, and our next research direction is how to reach a relative balance between resources utilization and resources workload in VM scheduling. In addition, since live migration does not support the size change of VMs in running, VM-Mig algorithm cannot calculate the changing sizes of the resources, so how to design this algorithm to adapt to the computing resource changes will be also our future concern.

ACKNOWLEDGEMENT

This work was supported by 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 Natural Science Foundation of Gansu Province, China (1308RJZA306); the Open Fund of the State Key Laboratory of Software Development Environment (SKLSDE- 2009KF-2-08)

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and Others, "Above the clouds: A berkeley view of cloud computing," Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley 2009.

[2] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: research problems in data center networks," *ACM SIGCOMM Computer Communication Review*, vol. 39, 2008, pp. 68--73.

[3] J. G. Koomey, "Estimating regional power consumption by servers: A technical note," *AMD Technical Study*, 2007.

[4] L. A. Barroso and U. Holzle, "The case for energy-proportional computing," *Computer*, vol. 40, 2007, pp. 33-37.

[5] "Amazon EC2,". http://www.idc.com

[6] A. Verma, P. Ahuja and A. Neogi, "pMapper: power and migration cost aware application placement in virtualized systems," in *Middleware 2008*: Springer, 2008, pp. 243-264.

[7] N. Bobroff, A. Kochut and K. Beaty, "Dynamic placement of virtual machines for managing sla violations," in *Integrated Network Management, 2007. IM'07. 10th IFIP/IEEE International Symposium on*, 2007, pp. 119-128.

[8] J. Dong, X. Jin, H. Wang, Y. Li, P. Zhang, and S. Cheng, "Energy-Saving Virtual Machine Placement in Cloud Data Centers," in Cluster Computing and the Grid, IEEE International Symposium on, 2013, pp. 618-624.

[9] M. Wang, X. Meng and L. Zhang, "Consolidating virtual machines with dynamic bandwidth demand in data centers," in INFOCOM, 2011 Proceedings IEEE, 2011, pp. 71-75.

[10] M. Al-Fares, A. Loukissas and A. Vahdat, "A scalable, commodity data center network architecture," in ACM SIGCOMM Computer Communication Review, 2008, pp. 63-74.

[11] X. Meng, V. Pappas and L. Zhang, "Improving the scalability of data center networks with traffic-aware virtual machine placement," in INFOCOM, 2010 Proceedings IEEE, 2010, pp. 1-9.

[12] V. Mann, A. Kumar, P. Dutta, and S. Kalyanaraman, "VMFlow: leveraging VM mobility to reduce network power costs in data centers," NETWORKING 2011, pp. 198-211.

[13] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2, 2005, pp. 273-286.

[14] E. G. Coffman Jr, M. R. Garey and D. S. Johnson, "Approximation algorithms for bin packing: A survey,", 1996, pp. 46--93.

[15] G. J. Woeginger, "Exact algorithms for NP-hard problems: A survey," in Combinatorial Optimization—Eureka, You Shrink!: Springer, 2003, pp. 185-207.

[16] Y. Shang, D. Li and M. Xu, "Energy-aware routing in data center network," in Proceedings of the first ACM SIGCOMM workshop on Green networking, 2010, pp. 1-8.

[17] A. Singh, M. Korupolu and D. Mohapatra, "Server-storage virtualization: integration and load balancing in data centers," in Proceedings of the 2008 ACM/IEEE conference on Supercomputing, 2008, p. 53.

[18] S. Chaisiri, B. Lee and D. Niyato, "Optimal virtual machine placement across multiple cloud providers," in Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific, 2009, pp. 103-110.
[19] T. C. Ferreto, M. A. S. Netto, R. N. Calheiros, and C. A. F. De Rose, "Server consolidation with migration control for virtualized data centers," Future Generation Computer Systems, vol. 27, 2011, p. 1027.
[20] X. Liao, H. Jin and H. Liu, "Towards a green cluster through dynamic remapping of virtual machines," Future Generation Computer Systems, vol. 28, 2012, pp. 469--477.
[21] W. Voorsluys, J. Broberg, S. Venugopal, and R. Buyya, "Cost of virtual machine live migration in clouds: A performance evaluation," in Cloud Computing: Springer, 2009, pp. 254-265.
[22] K. Deb, "Multi-objective optimization," Multi-objective optimization using evolutionary algorithms, 2001,pp. 13-46.
[23] R. Niranjan Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat, "PortLand: a scalable fault-tolerant layer 2 data center network fabric," in ACM SIGCOMM Computer Communication Review, 2009, pp. 39-50.
[24] T. Benson, A. Akella and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in Proceedings of the 10th annual conference on Internet measurement, 2010, pp. 267-280.
[25] D. S. Johnson, A. Demers, J. D. Ullman, M. R. Garey, and R. L. Graham, "Worst-case performance bounds for simple one-dimensional packing algorithms," SIAM Journal on Computing, vol. 3, 1974,p. 299.

Biographies

DONG Jiankang, received his M.S. degree from Peking University (PKU) in 2005, and is currently a Ph.D candidate of Beijing University of Posts and Telecommunications (BUPT). His research interests include virtualization technology and data center network. Email: dongjk @bupt.edu.cn.

WANG Hongbo, received the Ph.D. degree at the Beijing University of Posts and Telecommunications (BUPT) in 2006. Currently he is an Associate Professor in State Key Laboratory of Networking and Switching Technology of BUPT. His main research interests cover cloud computing, data center network, overlay network, and Internet measurement, in which he has published over 40 technical papers in referred journals and conference proceedings. Email: hbwang@bupt.edu.cn

LI Yangyang, received the B.E. degree in information engineering from Nanjing University of Information Science and Technology in 2009. He is currently working toward the Ph.D. degree in Computer Science at the Beijing University of Posts and Telecommunications. His current research interests include cloud computing and data center networks. Email: Liyangyang@live.com

CHENG Shiduan, Professor of Beijing University of Posts and Telecommunications (BUPT). Graduated at BUPT. In1980's and 1990's she twice joined Alcatel Bell, Belgium as a visiting scholar and involved in ISDN and ATM research work. From 1992 to 1999 she was the director of the National Laboratory of Switching Technology and Telecommunication Networks and a member of the steering committee of communications in national 863 program. Her current research interests are next generation Internet, QoS of Internet and data center networks. Email: chsd@bupt.edu.cn