Placement Algorithm in Virtual Machine Scheduling Based on Cloud Computing

Jun Lu*
Gansu Political Science and Law Institute, Lanzhou, China

*Corresponding author e-mail: 36060243@qq.com

Abstract. This paper mainly proposes a virtual machine placement algorithm, which uses the principle of graph segmentation. This algorithm can reduce the energy consumption of cloud services, Experimental simulation shows that compared with MBFD and TAVMP algorithm, this algorithm has a significant improvement in cup utilization, device energy consumption and SLA violation rate.

Key words: Placement, Graph Partitioning, Energy Consumption

1. Introduction
With the deep development of computer science, the rapid development of new-generation large-scale Internet applications enables the users to share computer resources through network to become reality. Combing with the advantages and characteristics of grid computing, distributed computing and parallel computing, and other technologies, the cloud computing intensively utilizes a considerable amount of low-priced resources stored in the data center clusters, consequently realizes resource sharing and collaborative work, and provides users with on-demand, flexible and reliable elastic services through network, greatly reducing the investment at the earlier stage and operation & maintenance cost at the later stage of the information system construction. In order to construct a technological environment satisfying a variety of service requirements, the cloud computing effectively integrates the distributed resources, and realizes the automatic decoupling of service usage and system management, and these characteristics enable the cloud computing resources to be highly utilized, but what corresponds to high energy consumption is a lower resource utilization rate[1]. Currently, the overall resource utilization rate of most cloud servers only occupies 10% - 50% of the total resources. During the development process of cloud computing, How to improve the utilization of the server is the key problem to be solved in the cloud server.

In the cloud computing environment, virtualization technology is mainly used to provide services to users. The implementation of virtual machine based technology enables the virtual machine scheduling to greatly impact the performance of cloud computing environment, including the utilization rate of physical machine, energy consumption of data center, and other aspects, and the implementation of one excellent virtual machine scheduling method may increase the resource utilization rate of system[2]. So how to effectively implement the virtual machine scheduling is a critical issue to decide the overall cloud performance and efficiency, and the research on virtual machine scheduling methods has a higher theoretical research meaning, as well as an important
The scheduling of virtual machine is divided into two aspects: placement and migration. This paper mainly researches the partitioning algorithm by the association graph in the placement process, and tests the superiority of the algorithm performance.

2. Virtual Machine Placement Model and Algorithm

In this paper, a graph segmentation algorithm is proposed, which reconstructs the association graph of virtual machine and adopts the improved k-path segmentation algorithm, so as to obtain various virtual machine subgroups with internal frequent communication, as well as a lower load correlation. In combination with the network topology in the cloud environment, the placement results of final virtual machine subgroups and internal individual virtual machines are determined by expanding the clusters[4].

2.1 Energy Consumption Model and Virtual Machine Model

The correlated virtual machine model is abstracted into an undirected graph K (V, M), and node V represents the individual virtual machine, while the boundary value f (KM, KM) represents the traffic between virtual machines. The attributes of each virtual machine include the demands on CPU and memory[5].

The energy consumption in the cloud computing environment mainly includes computer energy consumption and network equipment energy consumption. Computer energy consumption $E_{pm}$ is mainly related to the number of virtual machines and the varying load of the virtual machines. The energy consumption of a computer is basically linear with the CPU utilization.

$$w(u) = kW_{max} + (1 - k)W_{max}u$$ (1)

$$E_{pm} = \sum_{i=1}^{m} W_{pm}^i$$ (2)

In the equation above, $W_{max}$ represents the maximum power of physical machine at full capacity, k represents the energy consumption coefficient at the idle condition.

The switch consumes the most in the cloud server, which can refer to the energy consumption of the device. [6].

$$W_{sw} = W_{ch} + n_i W_{l} + \sum_{i=1}^{c} n_{ci} W_{ci}$$ (3)

$$W_{net} = \sum_{i=1}^{l} f_{sw}^i W_{sw}(t) dt$$ (4)

$W_{ch}$ represents the fixed power to maintain the start-up of switchboard, $W_{l}$ the power of each line card, $n_{i}$ the quantity of line cards plugged into the switchboard, C the quantity of different transmission rates, and $\sum_{i=1}^{c} n_{ci} W_{ci}$ the total energy consumption corresponding to the transmission rates. The energy consumption of network equipment can be reduced effectively only when the switchboards are activated as few as possible.

2.2 Virtual Machine Placement Algorithm for Graph Segmentation

This paper utilizes the partitioning method to partition the correlated virtual machine groups into various sub-groups, and ensure a greater traffic within the subgroups, a lower load correlation, and a smaller traffic among different subgroups[3]. Finally, in combination with the Fat-tree topology, the placement of individual virtual machines in the subgroups is completed by expanding the clusters. In a cloud data center with large-scale virtual machine clusters, the quantity and loads of virtual machines change constantly with the demands of applications and users. Therefore, when placing virtual machines, load dependence should be considered to avoid SLA violation. Placing virtual machines with high load dependence on the same computer will lead to disputes and SLA violations. The CPU utilization is a major factor impacting the task execution. This paper selects CPU as the computational
object of load correlation.

In this paper, the undirected graph $K'(V, M')$ is derived to obtain $K'(V, M')$, and the new boundary value of $p(VM_i, VM_j)$ in $K'(V, M')$ is determined by the load correlation between the boundary value and the virtual machine connected by the boundary[7].

$$p = \alpha \frac{f - \min(f)}{\max(f) - \min(f)} + \beta \frac{\delta - \min(\delta)}{\max(\delta) - \min(\delta)}$$  
(5)

The normalization treatment of load correlation $p$ and traffic by a method has eliminated the differences in the order of magnitude, $p(VM_i, VM_j)$ is obtained by the above equation, and the optimum value of $\alpha + \beta = 1$ is finally determined by experiments.

Next, the graph $K'(V, M')$ is partitioned. This paper proposes the multilayer K-path partitioning algorithm to partition the virtual machine association graph. The primal algorithm mainly includes three stages: after coarsening, initial partitioning and refinement of primal large-scale graphs, various sub-groups can be obtained. The communication within the subgroups is close, while the communication among subgroups is relatively sparse. The specific process is divided into the following stages[8]:

Coarsening: to convert primal graphs into sub-graphs, with the main purpose of reducing the scale of primal graphs and facilitating the partitioning at the next stage.

Initial partitioning: to initially partition the sub-graphs with a smaller scale obtained at the coarsening stage. After the coarsening stage, the scale of coarsened graphs obtained is smaller, and the partitioning of graphs with a smaller scale can effectively reduce the complexity[9].

Refinement: Mapping the segmentation results of subgraphs to detailed graphs. In the layer-by-layer mapping process, the minimum edge partition value is obtained by moving the nodes, and the optimal partitioning results of each layer are obtained by exchanging the nodes at the partitioning boundary. The refinement ensures that the partitioning results are the locally optimal solutions of each layer, so as to reach the optimal partitioning of primal graphs.

In this paper, NFCA (node fusion coarsening algorithm) is proposed to optimize each round of coarsening process in order to increase coarsening rate and accelerate coarsening process.  [4].

2.3 Resource Standardization

The virtual machine resource demands include CPU, RAM and other multi-dimensional resources, which is not conducive to the ordering of virtual machine entities. Therefore, this paper proposes a kind of resource standardization method to facilitate the resource ordering of virtual machine and physical machine entities. Firstly, the number of CPUs ($C_u$) and RAM ($R_u$) that appear most frequently in virtual machine resource requests are taken as resource units and the resources can be standardized on physical machine and virtual machine entities in accordance with the resource units[10], with the equation as below:

$$stand(PM_i) = \frac{pc_i}{C_u} + \frac{pm_i}{R_u}$$  
(6)

$$stand(VM_j) = \frac{vc_i}{C_u} + \frac{vm_i}{R_u}$$  
(7)

Wherein, initial value $PM_i(pc_i, pm_i) = (16 - core, 32GB)$, $VM_j(vc_i, vm_i) = (4 - core, 8GB)$, and standardized value $stand(PM_i) = 16$, $stand(VM_j) = 4$.

2.4 Placement of Virtual Machine Groups

Various sub-graphs with a smaller scale are obtained after partitioning of the virtual machine association graphs, and the communication within the sub-graphs is frequent while the communication among sub-graphs is sparse. Each subgraph represents a virtual machine subgroup. Placement is divided into two steps: 1. Virtual machine subgroup placement, that is, to determine the corresponding
relationship between virtual machine and computer cluster; 2. Virtual machine individual placement, that is, to determine the corresponding relationship between each virtual machine and computer in the cluster.

First, the resources are standardized and sorted in descending order for virtual machine subgroups by the resource standardization method to ensure the virtual machine subgroups with more resource demands are preferentially placed in the physical machine clusters which satisfy the placement conditions and have less residual resources. After traversing of all physical machine clusters, any virtual machine groups which haven't been placed are placed by expanding the clusters. Then, the physical machine clusters with the most residual resources are combined with the clusters under the switchboards at the same aggregation layer to expand the clusters. If the conditions still fail to be satisfied, the clusters under the switchboards at the same core layer are combined to expand the clusters[5].

2.5 Experiment Results
This paper utilizes CloudSim for simulation experiment, and validates the algorithm performance by comparison with MBFD algorithm and TAVMP algorithm from three aspects of CPU utilization, energy consumption and SAL violation rate. In the algorithm comparison, MBFD utilizes the greedy strategy to obtain the placement proposal of virtual machines with Low energy consumption by improving the best-fit decremental algorithm, and TAVMP also considers the relevance among virtual machines in the placement process. So this paper chooses the above algorithm to evaluate the performance of GPVMP.

(1) Setting of basic parameters
This paper carries out the experiment simulation by combing with the fat-tree network topology, and adopts three common scales in the cloud data center to conduct the experiment in order to compare the algorithm performance. The link capacity in the fat-tree structure is set from 1GB to 8GB. In the experiment, four kinds of physical machine examples corresponding to different capacities are set. If two virtual machines are related, Gaussian distribution is used to generate the transmission rate. The average value is set to 10Mbps, the standard deviation is set to 1Mbps, and the probability distribution is set to 0.75.

| Table 1. Setting of basic parameters |
|-------------------------------------|
| Configuration | Kernels | Memory (GB) | Power (W) |
| PM instan 1   | 16      | 32          | 370       |
| PM instan 2   | 16      | 16          | 370       |
| PM instan 3   | 8       | 16          | 300       |
| PM instan 4   | 4       | 8           | 250       |

(2) Result analysis
The CPU utilization in PlantLab is utilized for experiment. All data are sampled every 6 minutes, and 240 data will be obtained within 24 hours. The average CPU utilization is 0.44, so bound is set as 0.44. By placement of 600 virtual machines in Structure 1, the optimum value of $a$ is determined by comparing with the variations of total energy consumption under different settings of $a$. The value of $a$ is finally set as 0.7 to ensure low energy consumption and SLA violation rate.
Figure 1. Changes in total energy consumption corresponding to different $a$ values in Structure 1

(3) Resource utilization rate

Resource utilization is an important index to evaluate the performance of virtual machine placement algorithm, so this paper compares the average resource utilization of GPVMP, TAVMP and MBFD in three structures. The number of virtual machines placed is twice the number of physical machines under each structure to ensure high resource utilization. The experimental results are shown in Figure 2.

With the expansion of data scale, the utilization of CPU and memory resources are gradually reduced. GPVMP can obtain higher utilization of CPU and memory. MBFD effect is poor and causes more resource waste. TAVMP does not consider the resource fragmentation, resulting in relatively poor performance. Overall, GPVMP performs best, not only reducing resource fragmentation, but also improving the utilization of virtual machines.

Figure 2. Average utilization of CPU and RAM under three architectures

(4) Total energy consumption

It can be seen from Figure 3 that with the increase of virtual machines, the energy consumption of the three algorithms has increased, but the total energy consumption under gpvmp algorithm is the lowest, which shows that the algorithm can reduce the energy consumption of virtual machine placement.
Figure 3. Comparisons of Energy Consumption with the Number of Virtual Machines Increasing in Three Structures

(5) SLA violation rate
The results show that GPVMP has improved significantly in the controlling of SLA violation. The average SLA violation rate also increases along with the change of virtual machines. But the variation trends of SLA violation rates corresponding to different algorithms differ, as shown in Figure 4. GPVMP can control the growth of SLA violation rate in a small range, while the increment of SLA violation rates corresponding to the other two algorithms increases greatly. The average SLA violation rate of GPVMP is 29.8% lower than MBFD, and 19% lower than TAVMP.

Figure 4. SLA violation rate

3. Conclusions
In this chapter, GPVMP algorithm is proposed to solve the problem of high energy consumption in cloud computing. By introducing the load correlation of virtual machine, the load correlation is obtained in accordance with the CPU utilization computation to reflect the similarity of resource utilization of virtual machines. This paper conducts the partitioning of the virtual machine association graph in accordance with the load correlation and traffic, and proposes the NFCA algorithm to accelerate the partitioning and coarsening process. Meanwhile, the large-scale virtual machine association graphs are coarsened, and various virtual machine subgroups with an internal frequent communication, a lower load correlation, and sparse communication among the subgroups are obtained after initial partitioning and refinement. The resource ordering of virtual machine and physical machine entities is conducted in accordance with the resource standardization method, and the virtual machine subgroups and physical machine clusters, and the mapping relation between individual virtual machines in the subgroups and physical machine entities in the clusters are finally
determined by expanding the clusters. Finally, this paper validates the effectiveness of the proposed GPVMP algorithm. The experiment results show that GPVMP algorithm can ensure a higher resource utilization rate, effectively save the energy consumption of network equipment and that of physical machines in the data center, and effectively avoid the occurrence of SLA violation.

Acknowledgments
This work was financially supported by National natural science foundation of China (61562002) fund.

References
[1] chen Z G,DU K J,Zhan Z H,Deadline constrained cloud computing resources scheduling for cost optimization banse on dynamic objective genetic algorithm[C].//Evolutionary Computation,2015IEEE Congress on .New York:IEEEPress,2015:708-714
[2] Armbrust M,FOX A,Griffith R,etal.Avies of cloud computing [J].Communications of the ACM,2010,53(4):51-54
[3] Jia S,jing G,He P,etal.Efficient Algorithm for Energy-Aware Virtual Network Embedding[J].Tsinghua Science&Technology ,2016,21(4):407-413.
[4] Kumar KVP, Banga MK:Energy balanced dynamic cluster routing approach for WSN[a] IEEE international Conference on recent Trends in Electronics[C].Bangalore:IEEE RIEICT,216:141-147
[5] Hao F ,Kodialam M.Online allocation of virtual machines in a distributed cloud[J].IEEE/ACM Transactions on Networking,2017,25(1):238-241.
[6] Liu V,Lingshu LI,Tang H,etal.Hierarchical Coordination Strategy for vEPC Virtual Network embedding Based on Subgraph Isomorphism[J].Journal of Electronics&Information Technology,2017,39(5):1170-1176
[7] Sun L Q,Chen X Y.virtual network embedding based on computing, network and storage resource constraints[J].IEEE Internet of Things Journal,2017:1-1.
[8] Shahin A A. A Collective Neurodynamic Approach to Survivable Virtual Network Embedding[J].International Journalal of Advanced Computer Science&Applications,2018.9(3) :217-213
[9] Chowdhury S R ,et al. Dedicated Protection for Survivable Virtual Network Embedding[J]. IEEE Transactions on Network &Service Management,215,PP(99):36-44
[10] LIRA V,TAVARES E,et al.Dependable virtual network mapping[J].Computing,2015,97(5):459-481