Virtual Machine Live Migration Strategy in Big Data Information System

Juan Fang(✉), Lifu Zhou, and Mengxuan Wang

Faculty of Information Technology, Beijing University of Technology,
Beijing 100022, China
fangjuan@bjut.edu.cn

Abstract. In recent years, as an emerging technology, cloud computing has provided us with convenient services, and power consumption on issues have become increasingly prominent. Virtual machine live migration technology has become an important technology to reduce the power consumption of cloud computing centers. In the process of virtual machine migration, the performance of the virtual machine is inevitably degraded, which may violate service level agreement (SLA, Service Level Agreement). How to use virtual machine live migration technology to reduce power consumption as much as possible while ensuring a low SLA violation rate becomes a hot issue. This paper aims to optimize the light load detection and virtual machine redistribution in the virtual machine live migration model. Aiming at the problem that the existing virtual machine light load detection method is easy to cause "over-migration", this paper proposes a threshold-based minimum CPU utilization method for light load detection, which effectively avoids excessive virtual machine migration. Aiming at the problem that the current process of virtual machine re allocation algorithm is relatively simple, and there is a certain power loss space, we present power aware simulation annealing algorithm (PASA). The algorithm combines the simulated annealing algorithm based on the power aware best fit decreasing algorithm (PABFD), which largely avoids the disadvantage that the PABFD easily falls into the local optimal solution trap. The paper uses the CloudSim simulator as simulation platform. The results show that compared with the best algorithm combination proposed by the previous researchers, the power consumption of the new algorithm combination proposed in the paper is reduced by 16.79%, and the SLA violation rate is reduced by 85.37%. Combining the two algorithms together can lead to better energy efficiency, performance and quality of service than using the two algorithms.

Keywords: Cloud computing center · Virtual machine migration
Low load detection algorithm · Virtual machine reallocation algorithm

1 Introduction

The cloud computing concept emerged in 2006 and was developed on the basis of large-scale distributed computing technology. NIST defines cloud computing as a convenient model that can access the pool of computing resources on the network on demand and with less administrative effort [1]. The emergence of cloud computing

Z. Xu et al. (Eds.): Big Data 2018, CCIS 945, pp. 573–588, 2018.
https://doi.org/10.1007/978-981-13-2922-7_37

provides a more feasible way for general enterprises to meet their own processing data requirements. Compared with building their own data centers, the price is low, and there are advantages such as rapid implementation, low maintenance cost, and low IT staff demand [2]. For the above reasons, cloud computing has become a hot spot in the industry. While providing us with convenient services,

tasks submitted by users are usually big data computing tasks, the power consumption of cloud computing centers has become increasingly prominent. Virtual machine live migration technology in cloud computing center has become a practical and effective solution. Virtualization technology is the foundation of big data. Virtualization technology enables multiple virtual machines (VMs, virtual machines) to share the resources of the same physical machine in parallel as real physical machines, while ensuring isolation between virtual machines [3]. The live migration of the virtual machine can be used to adjust the load in the large cloud computing center, dynamically adjust the load according to the current load level and migrate all the virtual machines in the light-loaded physical node to other physical nodes with normal load. Finally, the emptied physical node is adjusted to sleep mode, thereby achieving the purpose of saving energy and reducing carbon emissions. However, in the process of virtual machine live migration, the user can only be as transparent as possible. In the migration process, the performance of the virtual machine is reduced, which seriously affects the user experience and may even violate SLA. Based on the above reasons, how to make better use of the advantages brought by virtual machine live migration while avoiding some drawbacks has become one of the most popular and most meaningful research contents.

There were outstanding contributions previously in the four modules of the virtual machine live migration - Over Overloading Detection, Host Underloading Detection, Virtual Forwarder Selection, and VM redistribution.

The purpose of overload detection is to determine whether a physical node is in an overload state. However, if a general detection method is used to determine whether the node is overloaded, it is likely that the physical node has entered an overload state before taking the corresponding measures after the determination is completed. Therefore, in recent years, the academic community has mainly used various methods to predict the impending overload. Kim et al. [4] proposed a local weighted regression method to determine Host Overloading Detection. The method is to apply the mathematical local weighted regression method to the overload detection. The core idea is to periodically collect the physical node load status data and fit the CPU utilization data for a period of time into a curve and use this curve to predict the CPU utilization for the next time period. The mathematical model of the method is relatively simple, and the calculation process is relatively simple, so it has a low time complexity. At the same time, it is also ideal in predicting the effect. This model is directly used in our experiment. Arianyan et al. [5] proposed the minimum migration time strategy, the strategy takes full account of the impact of virtual machine live migration on performance and service quality and can effectively avoid the rise of SLA violation rate but does not consider the problem of effectively reducing virtual machine load. Buyya et al. [6] proposed the maximum relational coefficient method to determine the VM migration selection problem. This method is able to determine whether a physical node in light load condition, but due to the process of safety factor need extra operation time, so the method may lead to detection of extra time overhead. To solve the Host Underloading Detection, Ferreto et al. [7] proposed the CPU arithmetic mean method. Virtual machine redistribution is to reallocate the migrated virtual machine to other physical nodes. After redistribution, on the one hand, it is required that the other physical nodes cannot be overloaded, and on the other hand, the energy consumption of all physical nodes is required to be as small as possible.

Beloglazov et al. [8, 9] proposed the optimal adaptive descending algorithm for energy perception, which is the application of the best adaptive descending algorithm in virtual machine redistribution. The general idea of the algorithm is to arrange all virtual machines in reverse order of CPU utilization, and then find out one physical node with sufficient resources in all physical nodes to carry the virtual machine and minimize the power consumption increment after migration. The experimental data of Beloglazov et al. show that the algorithm has lower energy consumption than the energy-aware first-time adaptive descending algorithm, but the algorithm structure and algorithm are relatively simple, and there is still room for further improvement in energy efficiency.

2 Host Underloading Detection Based on Threshold

2.1 Minimum CPU Utilization

The core idea of the Host Underloading Detection method is to periodically traverse all the hosts in the cloud data center, and to calculate the hardware usage of all the hosts. Then, the hosts are sorted according to the CPU utilization rate. Determine underloading host according to the inequality (1).

$$\mathrm{h} \in H | \forall a \in H, h_u \leq \mathrm{a}_u \tag{1}$$

Where h, a is a single host, H is the physical host list of the entire cloud data center, h_u is the CPU utilization of the h node, and a_u is the CPU utilization of the node.

The MCU (MCU, Minimum CPU Utilization) is one of the lowest power and SLA violation algorithms based on the results of a large number of experiments in [4]. But there is no scholar to propose a virtual machine live migration overhead. This gives the MCU a big flaw.

The results of the study [10] turned out that virtual machine live migration process will produce a certain power. On this basis, Zhou et al. [11, 12] proposed over-migration. Over-migration causes the VM moved to sleep mode when the host moved out. And this may lead to the situation described in (2).

$$P\mathrm{ower} < Power - Power_{saved} + Power_{migration} \tag{2}$$

Where Power is the total power consumption before the migration occurs. $power_{saved}$ is the power saved by adjusting some physical nodes to sleep mode after migration, and it is also the part we want to maximize. $Power_{migration}$ is the migration power overhead caused by moving some physical nodes out of VM. In the event of this, the migration of VMs is not worth the loss, because power consumption is even higher than before migration. Due to virtual machine live migration will cause a certain degree of performance degradation, which has violated the risk of SLA, so these unnecessary transfers can also lead to an increase in SLA violation. This is over-migration.

2.2 Minimum CPU Utilization Based on Threshold

Overview of the Algorithm

To solve the problem above, the paper proposes minimum CPU utilization based on threshold (MUT, Minimum Utilization with Threshold). The core idea of this method is to use a large number of experiments to find the best value of the threshold to make further restrictions on the determination of the original MCU on the light load. Determine whether the threshold of light load host CPU utilization as a constraint or not. If the threshold is lower than the threshold, it is determined that the host is currently in the light load state. If it is higher than the threshold value, it is determined that the load is normal. It is not difficult to predict that the algorithm can effectively avoid the over-migration problem mentioned in the previous section if the appropriate threshold is taken.

In summary, from the theoretical level of the algorithm has the following advantages:

1. Reduce unnecessary light load decisions that can lead to over-migration problems, thereby reducing unnecessary migration costs and performance degradation.
2. Compared with the original algorithm is likely to produce lower power and lower SLA violation.

Algorithm Structure

The specific flow of the MCU based on the threshold is to periodically traverse all the hosts in the cloud data center, statistics the hardware usage of all the hosts, and then sort the hosts according to the utilization rate of the CPU, and then determine the underloading according to the inequality (3).

$$\mathrm{h} \in H|\forall a \in H, h_u \leq a_u|h_u < threshold \tag{3}$$

Where h, a is a single host, H is the host list of the entire cloud data center, h_u is the CPU utilization of the h node, a_u is the CPU utilization of a node, and the threshold is the final threshold determined by the experiment.

If there is a host h, try to migrate all the VMs on the physical machine to other hosts without causing overload to other hosts. If it can be implemented, the VM on the host will be moved out of the scheme.

The pseudo code of MCUT is as follows (Table 1):

Table 1. pseudo code of MCUT

Algorithm1: Minimum CPU Utilization with Threshold

```
Input: host List, threshold   Output: feasibility of allocation
min Utilization ← MAX
allocated Host ← NULL
foreach host in host List do
    CPU Utilization ← host.getUtilization()
    if CPU Utilization < min Utilization then
        min Utilization← CPU Utilization
        allocated Host ← host
if allocated Host ≠ NULL then
    vm List ← getVMsFromHost(allocated Host)
    host List ← deleteHostFromList(host List, allocated Host)
    if min Utilization < threshold then
        if host List has enough resources for vm List then
            return True
    else
        return False
```

Algorithm Summary

Through the minimum CPU utilization method based on the threshold and the description of the specific algorithm structure above, the following judgment can be made:

1. After obtaining a suitable threshold, the threshold-based minimum CPU utilization method can effectively avoid the over-migration problem, so the focus of the follow-up work is to find a suitable threshold by a large number of experiments.
2. It can be inferred that in the process of finding the appropriate threshold, if the threshold is reduced from 100% (that is, for the Host Underloading Detection process does not make the second constraint) to 0% (that is, if all the host if the CPU utilization rate of 0%, it is determined that the load is normal, under normal circumstances this means that all host load is normal) in the process. Over-migration problem will gradually reduce or even disappear with the constraints of the gradual tightening. In the process the unnecessary power consumption due to over-migration is gradually reduced and the total power consumption is reduced. Then because of the tightening of CPU utilization limits, the normal light-load decision will be affected, and more and more hosts that are really in a light-load state will be judged to be normally loaded. The total power consumption will rise and the SLA violation rate is declining throughout this process because the virtual

machine live migration is decreasing which caused a performance degradation. Therefore, the optimal threshold is then taken at the minimum power consumption, where the over-migration problem is minimized due to threshold constraints and it do not affect the normal light load determination process.

3. After using the appropriate threshold, MUT is likely to be superior to the original CPU utilization method in both total power consumption and SLA violation rate.
4. In summary, the next section of the paper is about finding the appropriate threshold through a lot of scientific experiments and comparing it with the minimum CPU utilization method in terms of power consumption and SLA violation rate.

2.3 Experiments and Results Analysis

Experimental Design

The overall experimental idea is that building a simulation of the cloud computing center by using a cloud computing center simulator. In order to control the experimental variables, variables in this simulation of the cloud computing center will not be changed. After the simulation of the MCU and the other three matching algorithms in the simulation of the cloud computing center run the situation, which repeat 10 experiments. And determine the power consumption in this case and SLA violation data. Next, the Host Underloading Detection algorithm is replaced by a threshold-based MCU. The 20 sets of thresholds are taken from 100% to 0% of the difference. Each set of thresholds is repeated ten times. The power consumption and SLA violation data are also determined. And compare the power consumption under the optimal threshold value and SLA violation data with MCU experimental data. Finally, it proves that the proposed minimum CPU utilization method can be made improvements on the basis of predecessors.

Experimental Environment

This paper uses CloudSim 3.0.2 as a cloud computing center simulation platform. The simulator is one of the most powerful and powerful cloud computing platform simulators currently favored by researchers. The simulator is currently one of the most popular and most powerful cloud computing platform simulators. The simulator comes with an energy consumption and SLA violation rate monitoring module that automatically generates an operational report with these two data after each simulation run. In the experiment simulation of a 800 host with a medium-sized cloud computing center. Of which 50% of the host is Huawei Fusion Server Rh2288H. Each server is equipped with two Intel Xeon E5_2609 processors. The server model memory size holds 65G. Hard disk size holds 1 TB. Another 50% of the host model is equipped with the processing Switch to Intel Xeon E5_2699. The available bandwidth per host is 1Gbit/s.

Workload

In order to make the results of the simulation in this paper more realistic and effective, it is necessary to use the workload data of the real system environment, so we use some of the real data provided by the CoMon project. The specific workload data is the ten-day operational data randomly selected by PlanetLab from March to April 2011 recorded by the CoMon project. The specific data characteristics of the workload are shown in Table 2.

Table 2. Workload data characteristics (CPU utilization)

Date	Number of VMs	Average (%)	Sample estimation deviation
03/03/2011	1052	12.31	17.09
06/03/2011	898	11.44	16.83
09/03/2011	1061	10.70	15.57
22/03/2011	1516	9.26	12.78
25/03/2011	1078	10.56	14.14
03/04/2011	1463	12.39	16.55
09/04/2011	1358	11.12	15.09
11/04/2011	1233	11.56	15.07
12/04/2011	1054	11.54	15.15
20/04/2011	1033	10.43	15.21

Since the load data is derived from the real environment, and each group contains the entire plant running data of the entire PlantLab throughout the day, the user requests and tasks of different characteristics are evenly distributed among the ten groups of workloads. The experimental data is close to the real environment. Theoretically, it can be inferred that the experimental data obtained by applying these workloads and the conclusions based on experimental data are highly scalable.

Experimental Data and Results Analysis

Figure 1 is a graph of the final data obtained from the experiment for determining the threshold. The circular coordinate point uses the left ordinate for the power data and the X coordinate point for the SLA violation rate data using the right ordinate.

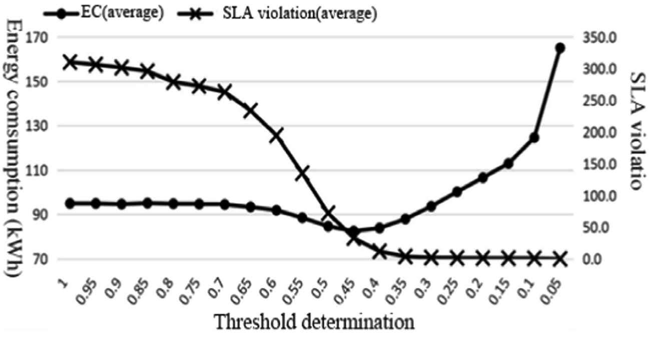

Fig. 1. Energy consumption and SLA violation with different threshold.

It is not difficult to see that the power at different thresholds and the SLA violation data are consistent with the predictions made in the previous section. As the threshold decreases, the power consumption decreases firs. And the minimum value is minimized when the over-migration problem is minimized. Then the power consumption is limited due to the normal host underloading detection. SLA violation is declining due to the gradual reduction of virtual machine live migration. When the threshold is 0.45, the power consumption is the minimum, and the SLA violation is at a relatively low point. The optimal threshold is 0.45.

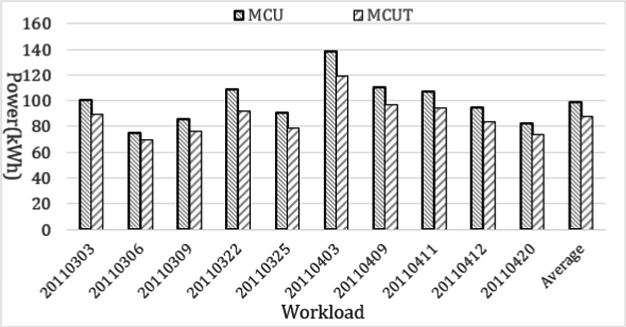

Fig. 2. Energy consumption and SLA violation with different threshold.

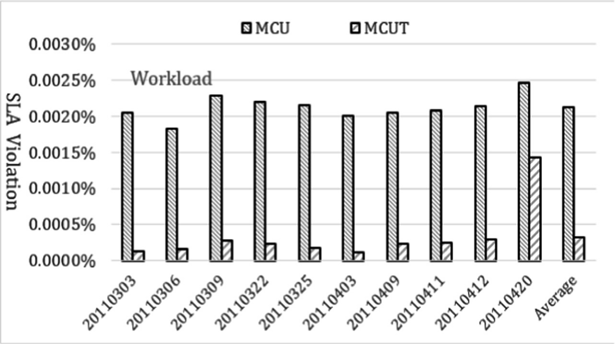

Fig. 3. SLA violation of two algorithms with different workloads

Figures 2 and 3 are compared with the algorithm of the minimum CPU utilization method when the threshold is 0.45. From the above two graphs, it can be seen that the minimum CPU utilization method based on threshold is significantly lower than the control group with the minimum CPU utilization method in terms of power consumption and SLA violation rate, especially in terms of SLA violation rate It is significantly reduced. From the average point of view, the algorithm proposed in this paper has a significant reduction in power consumption compared with the previous algorithm, the reduction rate is 11.88%, SLA violation is more obvious, the average decline of 84.47%. This shows that the threshold-based MCU proposed in this paper has improved the predecessor's algorithm and has made great progress.

3 Power Aware Simulation Annealing

3.1 Power Aware Best Fit Decreasing

VM reallocation is reassigned VM to other host up which host is identified as overload by Host Overloading Detection and egress selection module chooses to move out of the VM, and which host identified as light load through Host Underloading Detection. On the one hand, it is required that redistribution should not lead to the overload of the host. On the other hand, power of all the host moving in should increase as little as possible.

PABFD is a constructive heuristic algorithm to solve the packing problem. After a series of experiments, it is proved that the algorithm can give a smaller feasible solution in a tiny time complexity. But each time the allocation of VM only to find the a single VM optimal allocation of power under current circumstances, but not consider of the best overall direction.

For the above reasons, the paper aims to find an algorithm which can give a better solution than PABFD for the VM allocation algorithm. Because the VM reallocation process will increase the SLA violation at a certain extent if the waiting time is too long, which reduce the quality of service, so the new algorithm will control time complexity in a lower range.

3.2 Power Aware Simulation Annealing

I apply the simulation annealing algorithm to the VM reallocation module. Under the controlling of a cool down scheduler, the simulation annealing algorithm can be achieved in the acceptable time complexity. It can avoid falling into the local optimal "trap", so as to obtain a better perspective from the global point of view.

In this paper, PASA use solution given by PABFD as the initial solution. The state is recorded as the initial state i. According to the power model, we calculate the power of the entire cloud computing center, which recorded as E(i). Then we assign a VM randomly selected from the queue to be allocated to a randomly selected host with sufficient resources to carry the VM, which denote the state at that time as j. We calculate the whole power consumption of cloud computing center based on the power model, denoted by E (j). Calculate the power difference between the two states, denoted as ΔE. The specific formula as shown in (4).

$$\Delta E = E(j) - E(i) \tag{4}$$

If ΔE < 0, it is shown that the current solution provides a VM reallocation scheme with which power equal to or because of the current solution. So we accept j as a new solution; if ΔE > 0, it is shown that the solution is higher than the current solution power. However, for the possibility of jumping out of the local optimal solution trap, we will be accept this difference solution in a certain probability, where the probability is recorded as ζ. The specific probability formula as shown in (5).

$$\zeta = \exp(\frac{-\Delta E}{T}) \tag{5}$$

If $\zeta >$ random (0, 1), then accept the lower solution j as a new solution. If $\zeta \leq$ random (0, 1), then we give up j. And then continue to cool down and cycle the implementation of the steps until meet the termination conditions of cooling coefficient table set. Then calculate the entire cloud computing center power consumption of the final solution. And compare it with the power consumption of initial solution given by the PABFD. If the power consumption is lower than the initial solution, it is shown that PASA successfully found a better solution. Then we accept the final solution for the PASA VM reallocation program. If the power consumption is higher than initial solution, it is indicated that there requires more algorithm execution time to accept the solution. The power consumption is higher than PABFD, so we accept its initial solution as the final VM reallocation scheme.

3.3 Experimental Results and Analysis

The General Idea of the Experiment

There are three goals in this experiment. First, it is necessary to determine the value of each parameter in the cool down scheduler to obtain the best combination of parameters to achieve the best PASA effect. Then, in the same experimental environment, using the optimal combination of parameters obtained in the previous step, the most efficient one is selected from the three deployment scenarios through a large number of experiments as the final deployment plan. Finally, on the basis of a large amount of experimental data, we compare the power to SLA breaches by PABFD, which is aim to demonstrate the performance of PASA advantage through the mature algorithm proposed by the previous.

Due to the random search characteristics of the simulation annealing algorithm, the results given by PASA is likely to fluctuate within a certain interval. Therefore, all the experiments in this section will use a large number of experiments with taking the average of each sets of data to ensure the reliability of results.

Parameter Determination

To begin with the experiment, the parameters of the PASA are determined by using the parameter range of which is larger difference. In this experiment, the range of the initial temperature T is selected by {300, 600, 900}. The value of the iterations per time L is {200, 400, 600}. And the cooling coefficient β is in the range of {0.65, 0.8, 0.95}. Then we will choice 27 parameters randomly to combine with in every range of value. And take the average of ten experimental power and SLA violation data. The specific data is shown in Fig. 4.

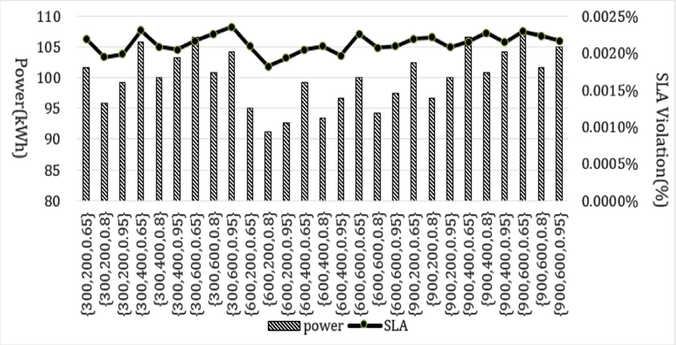

Fig. 4. Average power and SLA violation under different coarse-grained parameters

The combination of {600, 200, 0.8}, which is clearly visible, is the relative minimum of the power and SLA violation. The reference value for the next set of experiments is {600, 200, 0.8}. The specific value is based on the range of fine-grained parameters. The resulting average experimental data is shown in Fig. 5.

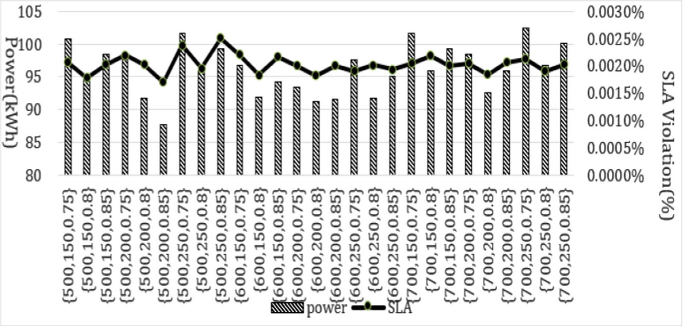

Fig. 5. Average power and SLA violation under different fine grain parameters

It can be seen from Fig. 5 that the combination of parameters {500, 200, 0.85} has the minimum average power and average SLA violation rate. There is a significant advantage over the other 26 sets of parameter combinations. The final combination is finalized {500, 200, 0.85}.

Experimental Results Analysis

It is significant to compare of the PASA and PABFD performance proposed in this paper. The reference index is still the power and SLA violation rate, which can measure the power efficiency of the two algorithms and the performance impact to the cloud computing center.

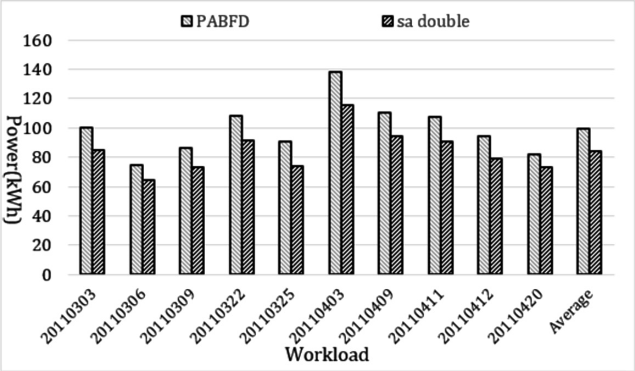

Fig. 6. Average power consumption of algorithms

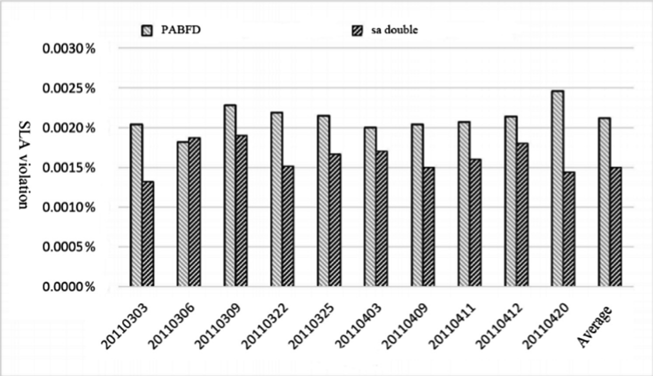

Fig. 7. Average SLA violation of algorithms

The specific experimental process is similar to the previous experiment. We test a number of times and take the average of the experimental results as the final reference data.

The specific power consumption data shown in Fig. 6. From the figure we can clearly see that the power under the method proposed PASA is lower than the control group by using PABFD. From the average of the 10 power groups under workloads, the algorithm proposed in this paper is 11.66% lower than that of the PABFD algorithm.

Figure 7 shows the SLA violation rate comparison of the two algorithms. It can be seen that under most workloads, the SLA violation of the energy-aware simulated annealing algorithm proposed in this paper is lower than PABFD. “20110306” is slightly higher under a set of workloads. From the average point of view, the average SLA violation rate of the energy-aware simulated annealing algorithm is 0.0015%, while the average SLA violation rate of PABFD is 0.0021%, which is 28.57% compared with the latter. The above data can be used to illustrate that the energy-aware simulated annealing algorithm is better than the previous ones in terms of performance and quality of service.

In summary, the PASA proposed in this paper is superior to the PABFD proposed by predecessors in the same experimental conditions, both in power and SLA violation, which is using the best combination of parameters obtained from the large number of experiments in the above two subsections. It can be explained that the algorithm proposed in this paper has some improvement on the basis of previous research in power efficiency and service quality.

4 Combination of Virtual Machine Live Migration Algorithm

4.1 Experimental Design

The overall experimental idea is to experiment with the same cloud computing center configuration and the same workload in the simulator. First, we get the power and SLA violation of the virtual machine live migration system using the threshold-based MCU and PASA matching algorithm. Then three groups of control data were obtained and compared with each other to demonstrate whether the two algorithms proposed in this paper can play the proper role in the same cloud computing center.

The first group of the three sets of algorithm combinations used as the control group is the best combination of a group of algorithms in the previous combination of algorithms. The second group of algorithms is a virtual machine based on the matching algorithm based on MCU. Live migration system, the third set of algorithm combinations is a separate use of PASA with matching algorithm composed of virtual machine live migration system. In order to guarantee the control variables in the experiment, the matching algorithm of the virtual machine live migration system proposed in this paper is consistent with the control group.

4.2 Experimental Configuration and Workload

Since the experimental process in the previous section is successful and there is no obvious problem and the resulting data is reliable, the experimental environment and the workload in this section are consistent with the experimental environment in chapter 3.4

4.3 Experimental Data and Analysis

We experiment multiple times with a set of target groups and three groups of control group algorithm combination. Taking the average of each group algorithm combination under each group of workload data.

Figure 8 is the combination of four algorithms of the power comparison. The figure of the four columnar data is legend from left to right, respectively, said the best proposed combination of the previous algorithm, a separate application based on the threshold of the MCU combination of algorithms, PABFD algorithm combination and combination of threshold-based MCU and PASA algorithm combination. From the final average data, we can see that the two algorithms combination proposed in this article decreases 16.70% compared to the previous combination of the best combination. It declines 7.36% compared to a separate application based on the threshold of the MCU power average. It decreases 5.70% compared to the average application of PABFD power alone.

Figure 9 for the combination of four algorithms SLA violation comparison. It can be seen that MCUT proposed to reduce over-migration and improve service quality is superior to PASA in performance and quality of service. In the combination of these two algorithms, the SLA violation rate is significantly lower than PASA alone and is basically the same as the control group applying MCUT.

From the specific data point of view, the proposed combination of the two algorithms in the average SLA violation compared to the previous combination of the best combination of the average algorithm SLA violation decreased by 84.43%. It compared to the average SLA violation of thresholds based on thresholds decreased by 3%. It

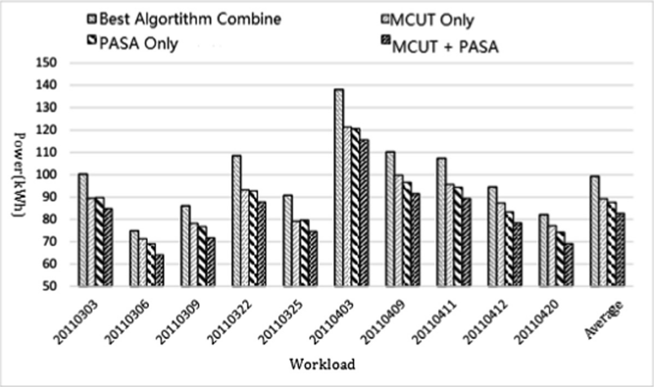

Fig. 8. Average power consumption of sets of algorithms

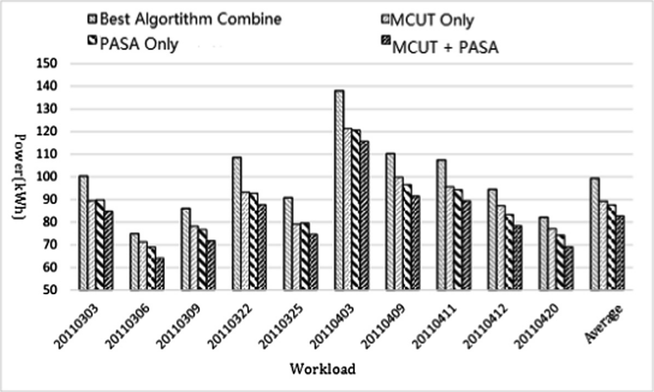

Fig. 9. Average SLA violation of sets of algorithms

compared with the average SLA violation of the control group with PASA alone reduced by 78.00%.

In summary, the two algorithms applied to a virtual machine live migration system giving full play to the threshold-based MCU which improves performance and quality of service features. The PASA reduce the power, and the combination of the two algorithms compared with the alone is a certain gain. There is a significant decline in the case of two algorithms at the same time power and SLA violation compared with the previous study of the best combination of the algorithm.

5 Conclusion

In this paper, we research on reducing power of virtual machine live migration system. Host Underloading Detection and VM reallocation module, which are related to the purpose, are the main module we aim to. We improved the new Host Underloading Detection method and VM Reallocation algorithm on the basis of previous research.

In the case of Host Underloading Detection, this paper proposes a minimum CPU utilization method disposed of judging light load carelessly when the minimum CPU utilization and leading to over-migration at a extent. This method limits the decision of the light load host by using an experimentally obtained optimal threshold as a quadratic constraint to ensure that only the host. That is really out of light load and does not raise the over-migration problem. The experimental data show that the proposed algorithm shows a significant reduction in power consumption compared with the previous algorithm. And the decrease rate is 11.88%. The SLA violation rate is more obvious, which decreases of 84.47% in average. It shows that the threshold-based minimum CPU utilization method proposed in this paper can effectively identify the light load state of the host. It can effectively avoid the occurrence of over-migration. On the basis of predecessors there is higher energy efficiency and better performance and quality of service.

In the case of VM reallocation, this paper presents PASA, which is a concrete application of simulation annealing algorithm VM reallocation. Using the VM reallocation scheme proposed by PABFD as the initial solution of the algorithm, the random change VM reallocation scheme adopts the new scheme. If the total power consumption of the cloud computing center is lower than the total power consumption of the initial solution, the total power dissipation is higher than the total power consumption of the initial solution. Then the new scheme is adopted with the probability of using the metropolis criterion. It can avoid the greedy algorithm in some ways which is easy to fall into the local optimal solution trap shortcomings. The experimental data show that the average power obtained by the algorithm proposed in this paper is 11.66% which is lower than that of PABFD and the average SLA violation rate is 28.57%. It shows that the PASA proposed in this paper can be better in ensuring performance and quality of service based on the lower power.

Finally, this paper demonstrates the effect of the two algorithms proposed in this paper in the same VM migration system. The experimental results show that the minimum CPU utilization method works well with PASA in the same cloud computing center. And both algorithms run at the same time compared to running one of the algorithms with better energy efficiency and better Performance and service quality.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (Grant No. 61202076), along with other government sponsors. The authors would like to thank the reviewers for their efforts and for providing helpful suggestions that have led to several important improvements in our work. We would also like to thank all teachers and students in our laboratory for helpful discussions.

References

1. Mell, P.: The NIST definition of cloud computing. Commun. ACM **53**(6), 50 (2011)
2. Yang, H., Tate, M.: A descriptive literature review and classification of cloud computing research. Commun. Assoc. Inf. Syst. **31**(2), 35–60 (2012)
3. Barham, P., Dragovic, B., Fraser, K., et al.: Xen and the art of virtualization. In: ACM SIGOPS Operating Systems Review, vol. 37, no. 5, pp. 164–177. ACM (2003)
4. Kim, N., Cho, J., Seo, E.: Energy-credit scheduler: an energy-aware virtual machine scheduler for cloud systems. In: Future Generation Computer Systems, pp. 128–137 (2014)
5. Arianyan, E.: Multi objective consolidation of virtual machines for green computing in Cloud data centers. In: 2016 8th International Symposium on Telecommunications (IST), pp. 654–659. IEEE (2016)
6. Buyya, R., Yeo, C.S., Venugopal, S., et al.: Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Futur. Gener. Comput. Syst. **25**(6), 599–616 (2009)
7. Ferreto, T.C., Netto, M.A.S., Calheiros, R.N., et al.: Server consolidation with migration control for virtualized data centers. Futur. Gener. Comput. Syst. **27**(8), 1027–1034 (2011)
8. 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. Concurr. Comput. Pract. Exp. **24**(13), 1397–1420 (2012)
9. Beloglazov, A., Abawajy, J., Buyya, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Futur. Gener. Comput. Syst. **28** (5), 755–768 (2012)
10. Strunk, A., Dargie, W.: Does live migration of virtual machines cost energy. In: Advanced Information Networking and Applications (AINA), pp. 514–521. IEEE (2013)
11. Fang, J., Zhou, L., Hao, X.: Energy and performance efficient underloading detection algorithm of virtual machines in cloud data centers. In: Cluster Computing (CLUSTER), pp. 134–135. IEEE (2016)
12. Sun, X., Ansari, N., Wang, R.: Optimizing resource utilization of a data center. IEEE Commun. Surv. Tutor. **18**(4), 2822–2846 (2016)