

Oriented-SLA and Energy-Efficient Virtual Machine Management Strategy of Cloud Data Centers

Lijun Xu¹, Weifeng Wang¹ and Xiaoqing Zhang^{2*}

¹*School of Computer and Information Engineering, Xinxiang University, Xinxiang, 453003, P.R.China*

²*School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan, 430023, P.R.China*

xljkeyan@126.com, 3104496134@qq.com, zxqtzy@126.com

Abstract

Cloud computing has revolutionized the information and communication technology industry by enabling on-demand provisioning of elastic computing resources on a pay-as-you-go basis. However, due to the large-scale popularization and application, cloud data centers consume enormous amounts of electrical energy, which results in high operating costs and carbon dioxide emissions. In this paper, twenty-two algorithms are simulated for testing the trade-off between the applications performance and energy consumption during virtual machine management of cloud data centers in CloudSim simulation toolkit. The goal of these algorithms is to reduce energy consumption under ensuring the performance of cloud applications. A large number of simulation experimental results prove the performance of these twenty-two algorithms respectively.

Keywords: *Virtual machine management, CloudSim, Service Level Agreement, Energy efficient*

1. Introduction

Cloud computing [1] is emerging as the latest distributed computing paradigm and attracts increasing interests of researchers in the field of distributed and parallel computing. The proliferation of cloud computing has resulted in the establishment of large-scale data centers containing thousands of computing nodes and consuming enormous amounts of electrical energy. The reason for this extremely high energy consumption is not just in the amount of computing resources used and the power inefficiency of hardware, but rather lies in the inefficient usage of these resources. High energy consumption by the infrastructure leads to substantial carbon dioxide emissions contributing to the greenhouse effect [2]. For reducing energy consumption in cloud data centers and ensuring SLA [3] between users and cloud resource providers, the paper constructs a simulation experiments researches to current existed oriented-SLA and energy efficient virtual machine management and placement strategies in [4], whose objective is to save energy and reduce the number of SLA violation.

¹For reducing energy consumption with meeting SLA, we split the problem of the adaptive placement of virtual machine into the following two parts:

- (1) Detection Strategy. Determining when a physical node is considered as being overloaded (or called hotspot) requiring migration one or more virtual machines from this node.

Xiaoqing Zhang is the corresponding author..

- (2) Selection Strategy. Selecting which virtual machines that should be migration from an overloaded physical server.

2. Related Works

Virtual machine placement in cloud data center is an important problem that remains to be effectively addressed. The mapping problem between virtual machines and physical hosts is to decide how to allocate virtualized resource on the cloud to many applications, thus is greatly impacts on the performance, cost and QoS guaranteed service. In ref.[5], a multi-objective optimization genetic algorithm is proposed to determine deployment strategy, which is subject to application SLA constraint. In ref.[6,7], the live migration problem of virtual machines is described as an optimization problem. The optimization objective is to minimize the energy consumption of physical hosts in cloud data center. But, the above work does not consider the performance and SLA violation from the view of user's applications. Hien Nguyen *et al.* [8] propose a resource management framework combining a utility-based dynamic virtual machine provisioning manager and a dynamic virtual machine placement manager. Both problems are modeled as constraint satisfaction problems. The virtual machine provisioning process aims at maximizing a global utility capturing both the performance of the hosted applications with regard to their SLA and the energy related operational cost of the cloud computing infrastructure. Different from Li Qiang *et al.* [5], Hien Nguyen *et al.* [8] lacks of the use of the live migration mechanism of virtual machines. R. Raghavendra *et al.* [9] have investigated the problem of power management for a data center environment by combining and coordinating five diverse power management policies. The authors explored the problem in terms of control theory and applied a feedback control loop to coordinate the controllers' actions. Sivadon Chaisiri *et al.* [10] propose an optimal virtual machine placement (OVMP) algorithm. The algorithm can minimize the cost spending in each plan for hosting virtual machines in a multiple cloud provider environment under future demand and price uncertainty.

Obviously, the virtual machine deployment problem in cloud computing environment mainly involves only a single target optimization at present. For example, the physical host used is least, or minimizing SLA violations and guaranteeing QoS performance of applications, or reducing the total energy consumption of data center and virtual machine migration number. However, these optimized goals may be in conflict with each other. In order to reduce the number of hosts used, all virtual machines need to be deployed reasonably to a smaller number of hosts in bin-packing mode, which can reduce the overhead of idle nodes. But this needs multiple virtual machine migrations to achieve. In order to guarantee the QoS of applications and reduce SLA violation, virtual machines have to be consolidated and migrated in many times, which can turn on more hosts and lead to more energy consumption. Meanwhile, the energy consumption is associated with the specific resource utilization of physical hosts, so the least used hosts can not always bring the minimum energy consumption. An efficient virtual machine deployment strategy should balance and compromise the energy consumption, QoS and SLA.

3. Host Overload Detection Algorithms-Detection Strategy

3.1 MPA

The main idea of MPA (Maximal Power Algorithm) is known as that all hosts consume maximum power all the time in a heterogeneous non-power aware cloud data center, which means the CPU keeps the maximal power condition.

3.2 DVFS

DVFS is also called as Dynamic Voltage and Frequency Scaling [11]. The main idea of DVFS is: when CPU is not fully utilized, the performance of CPU can be reduced through reducing the supply voltage and clock frequency of CPU, which can reduce the dynamic power consumption with cubic orders of magnitude and can not affect the performance.

3.3 STH

STH is called as Static Threshold. It is based on the idea of setting upper utilization thresholds for nodes and keeping the total utilization of the CPU by all the virtual machines under the threshold. If the CPU utilization of a physical node exceeds the static upper threshold, some virtual machines have to be migrated from the node to reduce the CPU utilization in order to prevent a potential SLV violation, as showed in Figure 1.

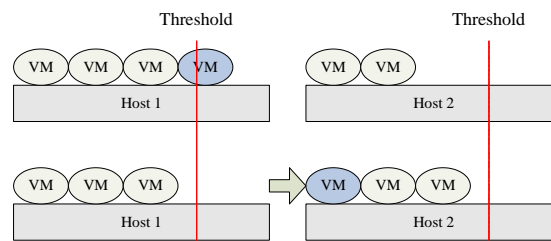


Figure 1. STH

3.4 IQR

Interquartile Range (IQR) is a measure of statistical dispersion. It is equal to the difference between the third and first quartiles: $IQR=Q_3-Q_1$. Unlike the total range, the interquartile range is a robust statistic, having a breakdown point of 25%, and thus is often preferred to the total range. In the overload algorithm based on IQR, the CPU utilization threshold is defined as,

$$T_u = 1 - \varepsilon \times IQR \quad (1)$$

Where ε is a parameter of the method defining how strongly the system tolerates host overloads. In other words, the parameter allows the adjustment of the safety of the method: a lower value of ε results in a higher tolerance to variation in the CPU utilization, while possibly increasing the level of SLA violations caused by the consolidation.

3.5 LR

The main idea of the Local Regression method [12] is fitting simple models to localized subsets of data to build up a curve that approximates the original data. The observation (x_i, y_i) are assigned neighborhood weights using the tricube weight function shown in Equation (2).

$$T_u = \begin{cases} 0, & \text{otherwise} \\ (1 - |u|^\beta)^3, & \text{if } |u| < 1 \end{cases} \quad (2)$$

If the inequalities (3) are satisfied, the algorithm detects a host overload, requiring some VMs to be offloaded from the host.

$$\begin{cases} \varepsilon \cdot \hat{y}_{i+1} \geq 1 \\ x_{k+1} - x_k \leq t_m \end{cases} \quad (3)$$

Where t_m is the maximum time required for a migration of any of the VMs allocated to the host.

3.6 RLWR

RLWR can adjust the value of the static upper utilization threshold based on a statistical analysis of historical data collected during the lifetime of virtual machines. Based on LR, the fit is evaluated at the x_i to get the fitted value \hat{y}_i , and the residuals $\hat{e}_i = y_i - \hat{y}_i$. At the next step, each observation (x_i, y_i) is assigned an additional robustness weight r_i , whose value depends on the magnitude of \hat{e}_i . Each observation is assigned the weight $r_i w_i(x)$, where r_i is defined as,

$$r_i = B(\hat{e}_i / 6s) \quad (4)$$

Where $B(u)$ is the bisquare weight function.

$$B(u) = \begin{cases} 0, & \text{otherwise} \\ (1 - |u|^2)^2, & \text{if } |u| < 1 \end{cases}, s = \text{median } |\hat{e}_i| \quad (5)$$

If the inequalities (6) are satisfied, the algorithm detects a host overload, requiring some VMs to be offloaded from the host.

$$\begin{cases} \varepsilon \cdot \hat{y}_{i+1} \geq 1 \\ x_{k+1} - x_k \leq t_m \end{cases} \quad (6)$$

Where t_m is the maximum time required for a migration of any of the VMs allocated to the host.

3.7 MAD

Median Absolute Deviation (MAD) is a robust statistic, being more resilient to outliers in a data set than the standard deviation. In standard deviation, the distances from the mean are squared leading to large deviations being on average weighted more heavily. For a univariate data set x_1, x_2, \dots, x_n , MAD is defined as the median of the absolute deviations from the median of the data set:

$$MAD = \text{median}_i (|x_i - \text{median}_j (x_j)|) \quad (7)$$

In the overload detection algorithm based on *MAD*, CPU utilization threshold T_u is defined as showed in the following,

$$T_u = 1 - \varepsilon \times MAD \quad (8)$$

4. Virtual Machine Selection Algorithms-Selection Strategy

4.1 MPM

MPM (Minimum Period Migration) migrates a virtual machine v that requires the minimum period to complete a migration relatively to the other virtual machines allocated to the node. The migration period is defined as the amount of *RAM* utilized by

the virtual machine divided by the spare network bandwidth available for the physical node j . Let V_j be a set of virtual machines currently allocated to the node j . The set R of migrated virtual machines searched by MPM is

$$\begin{aligned} & \min \sum_{v \in R} RAM_u(v) / NET_j \\ & s.t. \begin{cases} R \in Q(V_j) \\ u_j - \sum_{v \in R} u_a(v) < T_u \\ Q(V_j) = \{U \mid U \subseteq V_j\} \end{cases} \end{aligned} \quad (9)$$

Where, $RAM_u(v)$ is the amount of RAM currently utilized by the virtual machine v and NET_j is the spare network bandwidth available for the node j .

4.2 MNM

MNM (Minimum Number Migration) selects the minimum number of virtual machines needed to migrate from a physical node to lower the CPU utilization below the upper utilization threshold if the upper threshold is violated. Let V_j be a set of virtual machines currently allocated to the node j , $Q(V_j)$ is the power set of V_j (include null set and all subset of universal set). The set R of migrated virtual machines searched by MNM is

$$\begin{aligned} & \min |R| \\ & s.t. \begin{cases} R \in Q(V_j) \\ u_j - \sum_{v \in R} u_a(v) < T_u \\ Q(V_j) = \{U \mid U \subseteq V_j\} \end{cases} \end{aligned} \quad (10)$$

Where, u_j is the current CPU utilization of the node j , $u_a(v)$ is the fraction of the CPU utilization allocated to the virtual machine v .

4.3 MU

MU (Minimum Utilization) prefers to migrate the virtual machines with the lowest CPU utilization. MU is suit for the virtual machine placement strategy with priority. The virtual machine with high CPU utilization has high priority and the virtual machine with low CPU utilization has low priority.

4.4 RCA

RCA (Random Choice Algorithm) depends on a random selection of a number of virtual machines needed to reduce the CPU utilization by a physical node below the fixed utilization threshold. The set R of virtual machines find by RCA:

$$R \leftarrow X \stackrel{d}{=} U(0, |Q(V_j)| - 1) \quad (11)$$

X is a uniformly distributed discrete random variable, whose value index subset of V_j .

5. Simulation Experiments

CloudSim [13] has been selected as a simulation platform, as it is a modern simulation framework aimed at cloud computing environment. Contrasting to other simulation toolkits, such as SimGrid and GangSim, it allows the modeling of virtualized environments, supporting on-demand resource provision and their management. It has been extended to enable energy aware simulations, as the core framework does not provide this capacity.

The cloud environment includes a data center, which consists of N heterogeneous physical nodes. According to different process power of CPU, these physical nodes are divided equally into two types, represented by (1800MIPS, 2600MIPS). Two types of physical nodes both include two process elements (CPU core), 4G RAM and 1GB/s network bandwidth. Each virtual machine requires one CPU core with 2500, 2000, 1500, 500MIPS, 0.8, 1.7, 1.5, 0.6BG of RAM and 100MB of bandwidth. Each virtual machine runs a web-application or any kind of application with variable workload, which is modeled to generate the utilization of CPU according to a uniformly distributed random variable. The application runs for 150,000 MI that is equal to 10 min of the execution on 250 MIPS CPU with 100% utilization. Initially, all virtual machines are allocated according to the requested characteristics assuming 100% CPU utilization.

In order to compare the efficiency of the algorithms we use the following eight metrics to evaluate their performance.

- (1) Total energy consumption. The total energy consumption of all physical hosts in data center during the execution of application loads.
- (2) The number of virtual machine migration. The number of virtual machine migration during the auto-adaptive virtual machine optimization placement and consolidation.
- (3) The SLAV time per active host (SLATAH)

It is a percentage of time, during which active hosts have experienced the CPU utilization of 100%.

$$SLATAH = \frac{1}{N} \sum_{i=1}^N \frac{T_{i,s}}{T_{i,a}} \quad (12)$$

- (4) The ratio of performance degradation (RPD)

It means the overall performance degradation by virtual machines due to migrations.

$$RPD = \frac{1}{M} \sum_{i=1}^M \frac{C_{i,d}}{C_{i,t}} \quad (13)$$

Where N is the number of physical hosts, $T_{i,s}$ is the total time during which the physical host i has experienced the utilization of 100% leading to a SLAV, $T_{i,a}$ is the total of the physical host i being the active state (serving virtual machines), M is the number of virtual machines, $C_{i,d}$ is the estimate of the performance degradation of the virtual machine j caused by migration, and $C_{i,t}$ is the total CPU capacity requested by the virtual machine j during its lifetime. In our experiments, we estimate $C_{i,d}$ as 10% of the CPU utilization in MIPS during all migrations of the virtual machine j .

- (5) The service ratio of single node. It is the product of RPD and SLATAH.
- (6) The overall SLA violation. Overall SLA violation (SLAV) is defined as a fraction of the difference between the requested MIPS by all the virtual machines $U_{j,r}(t)$ and the actually allocated MIPS $U_{j,a}(t)$ relatively to the total requested MIPS over the lifetime of the virtual machines, as showed in following equation,

$$SLAV = \frac{\sum_{j=1}^M \int_t [U_{j,r}(t) - U_{j,a}(t)] dt}{\sum_{j=1}^M \int_t U_{j,r}(t) dt} \times 100\% \quad (14)$$

Where, M is the number of virtual machines. This metric represents the percentage of the CPU performance that has not been allocated when demanded by application relatively to the total demand.

- (7) The average SLA violation. The proportion of not allocated CPU to the total requirement brought by virtual machine migration.
- (8) The number of shutdown. The number of shutdown during the auto-adaptive virtual machine optimization placement and consolidation.

The following eight figures (Figure 2~9) shows that the changes of the performance metric of algorithms under different placement scale of virtual machines. The placement scale of virtual machines is defined as the number of physical nodes and virtual machines. Let the number of virtual machines equals to the number of physical nodes.

Figure 10 is to show the effect of different thresholds on the energy consumption and SLA of STH. with the increase of the fixed threshold, the energy consumption reduces but overall SLAV increases. This is because the increasing threshold will lead to more virtual machine consolidation, which brings less virtual machine migration. From the above experimental results, we can see the greater threshold dose not mean better, which is related to the tradeoff between the energy consumption of hosts and the performance of service.

Figure 11 is to show the effect of different safe factors on the energy consumption and SLA of RLWR, LR, MAD and IQR. We can see that with the increase of the safe factor, the energy consumption increases, the overall SLAV reduces and the number of virtual machine migration increases. This is because the safe factor ϵ is a parameter of the method that defines how aggressively the system consolidates virtual machines. In other words, the lower the safe factor ϵ , the less the energy consumption, but the higher the level of SLAV caused by the consolidation.

Figure 12 is to show the effect of different smoothness degrees on the energy consumption and SLA of RLWR and LR. The smoothness factor does not greatly effect on the energy consumption of the algorithm. But, the overall SLAV increases and the number of virtual machine migration drops off. This is because the increase of smoothness factor demonstrates the number of observation points increases for estimating a point, which makes the algorithm more robust in terms of bearing load peaks. As a result, this will bring less virtual machine migration and more SLAVs.

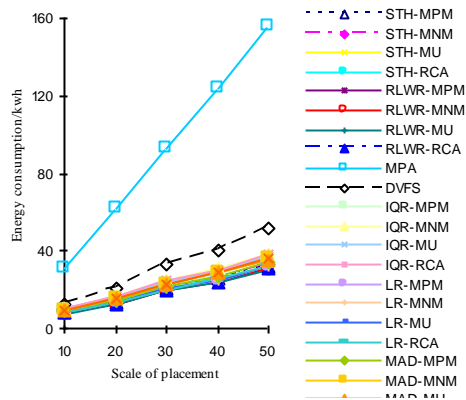


Figure 2. Energy Consumption

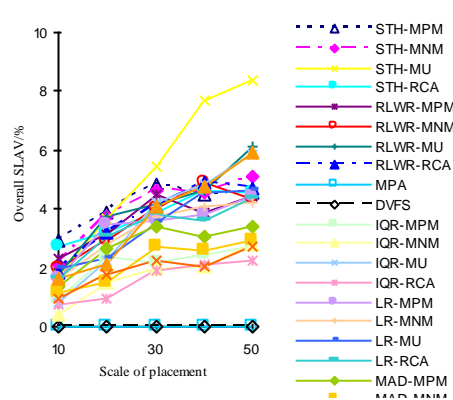
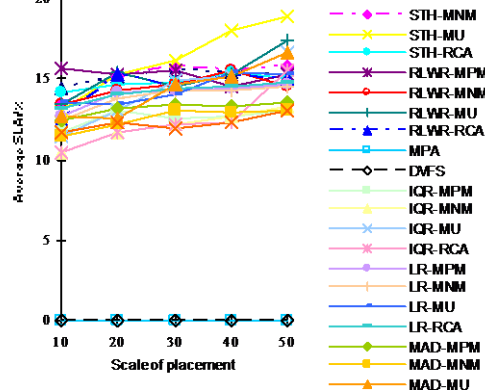
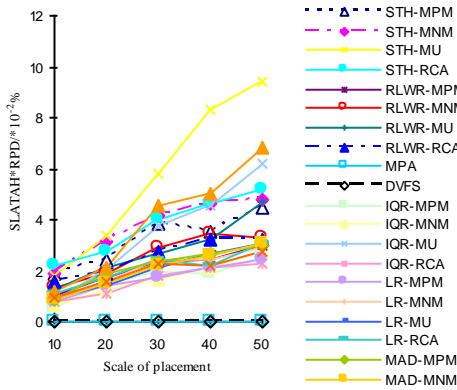
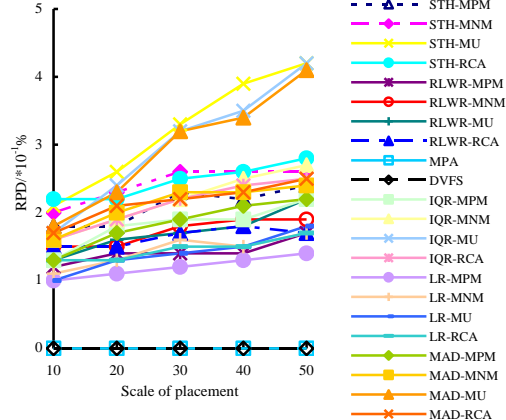
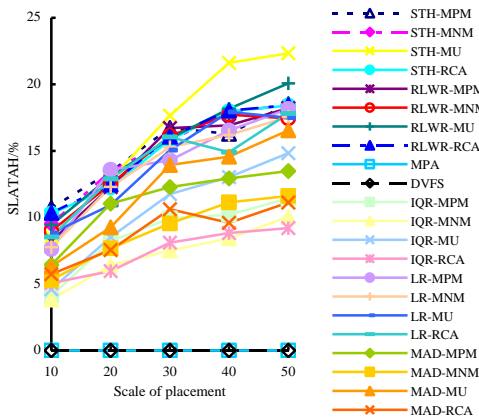
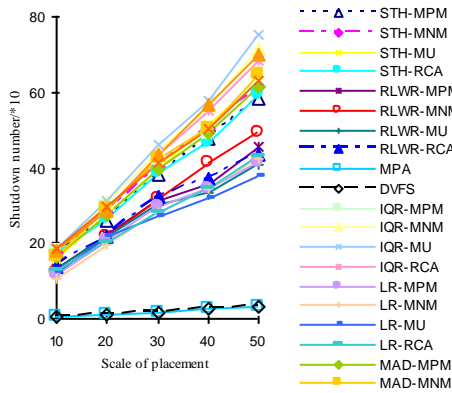
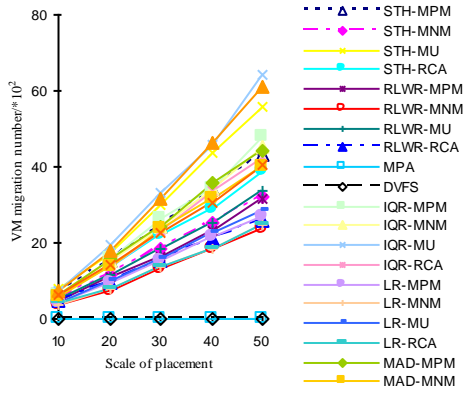


Figure 3. Overall SLAV



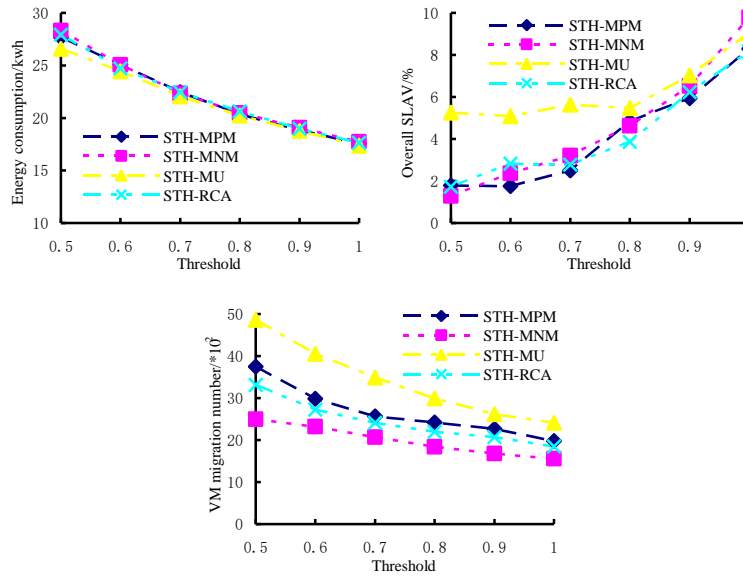


Figure 10. Threshold Effects on STH's Performance

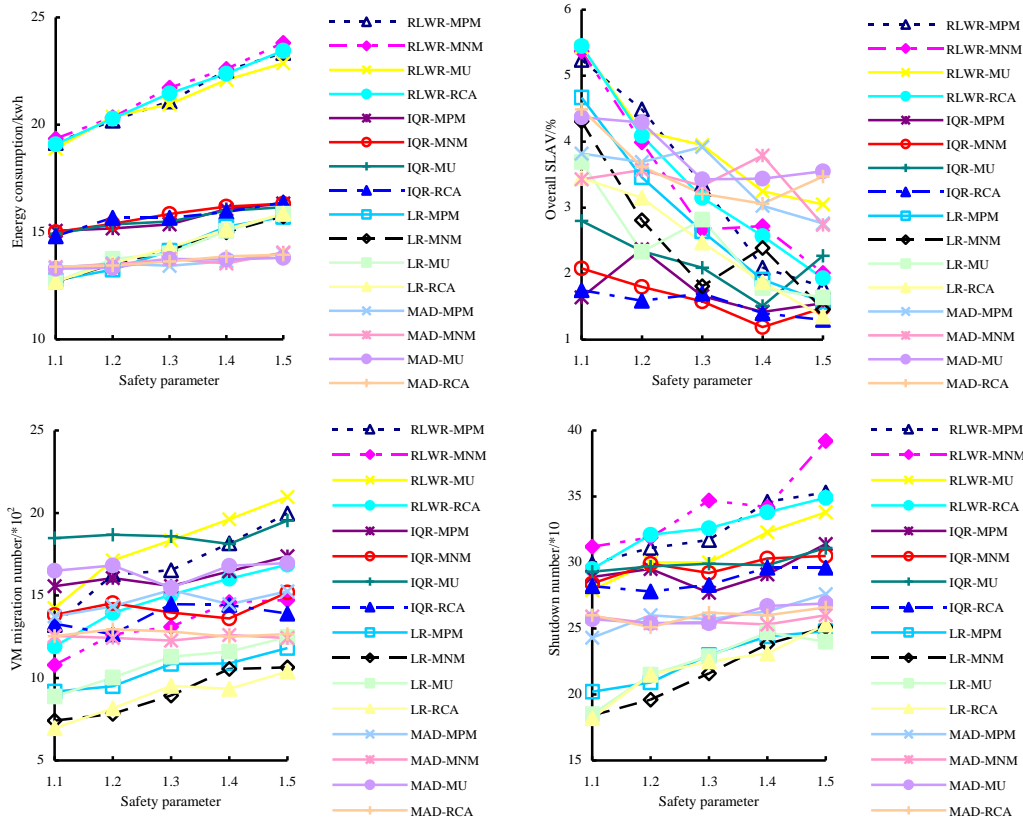


Figure 11. Safety parameter Effects on the Performance of RLWR, IQR, LR and MAD

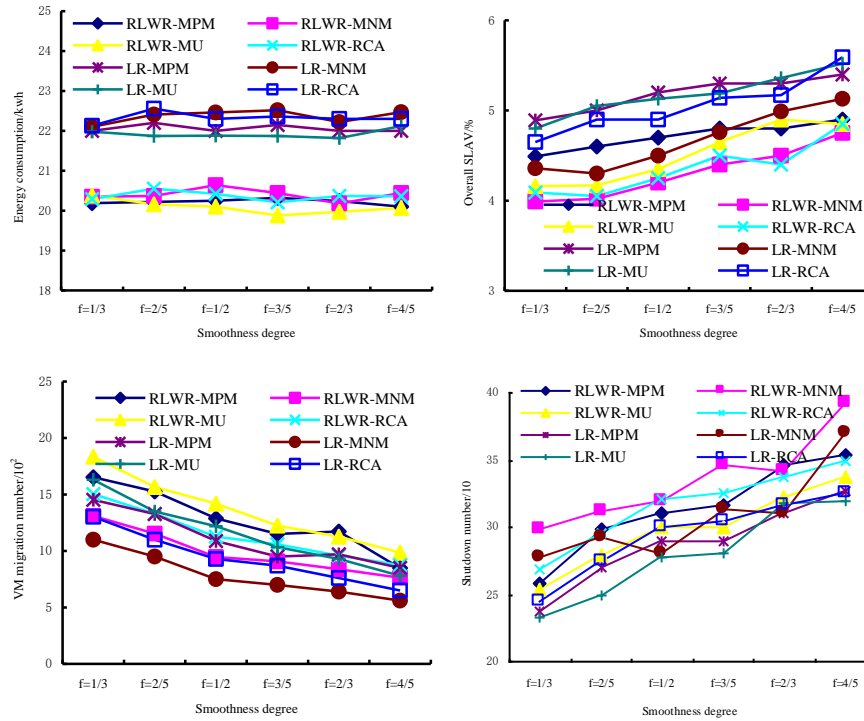


Figure 12. Smoothness Degree Effects on the Performance of RLWR and LR

6. Conclusion

Oriented-SLA and energy-efficient virtual machine management strategies of cloud data centers are researched in this paper through a number of simulation experiments. We can see that virtual machines are not only deployed on fewer hosts in these self-adaptive placement strategies, which promotes energy efficiency through turning off unused hosts, but also the load prediction of resource can bring high-reliable QoS delivery and avoid overmuch SLA violations. Experimental results show that these strategies have a good effect on decreasing SLA violation under ensuring energy efficiency.

Acknowledgements

This work was financed partially by the project of department of education in Henan province (15A520094).

References

- [1] L.M. Vaquero, L. Rodero-Merino, J. Caceres, "A break in the clouds: towards a cloud definition", SIGCOMM Computer Communication Review, vol. 39, (2008), pp.50-55.
- [2] A. Beloglazov, J. Abawajy, R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing", Future Generation Computer Systems, vol.18, (2012), pp.755-768.
- [3] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, I. Brandic, "Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility", Future Gener Comput System, vol.25, (2009), pp.599-616.
- [4] A. Beloglazov, R. Buyya, "Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers", Concurrency and Computation: Practice and Experience, vol.24, (2012), pp.1397-1420.

- [5] L. Qiang, H. Qin-Fen, X. Li-Min, "Adaptive management and multi-objective optimization for virtual machine placement in cloud computing", Chinese Journal of computers, vol.34, (2011), pp.2253-2264.
- [6] W. Xiaorui, W. Yefu, "Coordinating Power Control and Performance Management for Virtualized Server Clusters", IEEE Transactions on Parallel and Distributed Systems, vol. 22, (2010), pp.245-259.
- [7] J. Gueyoung, M.A. Hiltunen, K.R. Joshi, "Mistral: Dynamically Managing Power, Performance, and Adaptation Cost in Cloud Infrastructures", 2010 IEEE 30th International Conference on Distributed Computing Systems (ICDCS),(2010).
- [8] H. Nguyen, V., F.D. Tran, J.M. Menaud, "Performance and Power Management for Cloud Infrastructures", 2010 IEEE 3rd International Conference on Cloud Computing (CLOUD), (2010), pp.329-336.
- [9] R. Raghavendra, P. Ranganathan, V. Talwar, Z. Wang, X. Zhu, "No "power" struggles: coordinated multi-level power management for the data center", SIGARCH Computer Architecture News, vol.36, (2008), pp. 48-59.
- [10] S. Chaisiri, B-S Lee, D Niyato, "Optimal virtual machine placement across multiple cloud providers", Proceedings of APSCC'2009, (2009), pp.103-110.
- [11] G., Magklis, G. Semeraro, D.H. Albonese, "Dynamic frequency and voltage scaling for a multiple-clock-domain microprocessor", IEEE Micro, vol.23, (2003), pp.62-68.
- [12] W.S. Cleveland, "Robust locally weighted regression and smoothing scatterplots", Journal of the American Statistical Association, vol.74, (1979), pp.829-836.
- [13] R. Buyya, R. Ranjan, R.N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities", International Conference on High Performance Computing & Simulation, (2009), pp.1-11.

Authors



Lijun Xu, He is currently a Ph.D. student in the School of Computer Science and Technology at Wuhan University of Technology and an associate professor at the Xinxiang University. He received his B.S. from Henan Normal University, Xinxiang, Henan,China, in 2001, and M.S. from the Wuhan University of Technology, Wuhan, Hubei, China, in 2007. He has worked at the Xinxiang University since 2001. His research interests include resource management in cloud computing environments and Computer network. He has published about 10 papers.



Xiaoqing Zhang, He is a lecturer of Wuhan Polytechnic University. He received an M.S. degree in Computer Science and Technology and the Ph.D. degree in Computer Application Technology from Wuhan University of Technology. His current research interests include cloud computing, the Internet of things and wireless sensor networks.

