

Research Article

FP-ABC: Fast and Parallel ABC Based Energy-Efficiency Live VM Allocation Policy in Data Centers

Jianhua Jiang,^{1,2,3} Yunzhao Feng,^{1,3} Milan Parmar,⁴ and Keqin Li⁵

¹School of Management Science and Information Engineering, Jilin University of Finance and Economics, Changchun 130117, China

²Laboratory of Logistics Industry Economy and Intelligent Logistics, Jilin University of Finance and Economics, Changchun 130117, China

³Jilin Province Key Laboratory of Internet Finance, Jilin University of Finance and Economics, Changchun 130117, China

⁴International Department, Faculty of Computer Science and Information Management, Hunan University of Arts and Science, Changde, Hunan 415000, China

⁵Department of Computer Science, State University of New York, New Paltz, NY 12561, USA

Correspondence should be addressed to Keqin Li; lik@newpaltz.edu

Received 23 March 2016; Accepted 16 June 2016

Academic Editor: Fabrizio Riguzzi

Copyright © 2016 Jianhua Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Virtual machine (VM) technology is one of the energy-efficiency approaches to save energy with acceptable quality of service (QoS). In our previous studies, Artificial Bee Colony (ABC) based VM allocation policy can make a good tradeoff between performance and energy consumption. However, there are two problems in state-of-the-art ABC based approaches: (1) how to find global optimized solutions efficiently; (2) how to minimize the decision time of VM allocation. To solve these two problems, the idea of simulated annealing is adopted to get a better global optimum, and the idea of gradient descent is applied to accelerate the speed of finding solution space in Δt . Compared with state-of-the-art ABC based policies, the experimental results show that the proposed algorithm efficiently reduces energy consumption and SLA violation.

1. Introduction

At present, a large number of data centers have been put into practice to support internet-based services. Many applications, such as E-commerce and scientific computing, require large-scale data-intensive computing. As data centers and their applications continue increasing, how to realize energy-efficiency computing has become a particular challenge.

There are several properties of data centers making the problem of energy-efficiency computing difficult to be solved. First, workload of a data center is a priori unknown to energy-efficiency policy and will likely be variable over both time and space. As a result, a static energy-efficiency policy is insufficient. Second, customers wish their jobs can be finished in time with an acceptable price. Third, lots of computing and storage nodes are occupied in finishing their jobs in time with high energy consumption and low utilization.

Therefore, finding a tradeoff between energy consumption and performance in data centers is an important issue.

Many practices have been applied to achieve energy-efficiency, such as Dynamic Voltage and Frequency Scaling (DVFS) [1] and virtual machine technique [2]. Virtual machine technique can improve the utilization rate of resources, which are computing and storage nodes. For example, low-load nodes will be slept to save energy when their VMs have been migrated to other nodes with workload balance principles.

VM allocation policies with workload balance are considered as one of the important energy-efficiency solutions in data centers. In CloudSim [3] simulator, the classical policies have been proposed, such as Random Selection (RS), Minimum Migration Time (MMT), and Median Absolute Deviation (MAD), to make VM allocation decisions. Zhao et al. [4] applied PSO algorithm to VM allocation

policy, and they have achieved significantly on the measurement of energy-efficiency and QoS. In our previous works, Artificial Bee Colony (ABC) [5] is adopted to achieve energy-efficiency [6]. However, ABC based VM allocation policies have disadvantages of finding global solutions in time.

To solve these problems, the idea of simulated annealing is adopted to find the global solution as much as possible in this paper. Based on the principle of simulated annealing, following bees have probability to jump out of local optimum to find global optimum in other solution space. Even more, the idea of gradient descent is applied to accelerate the speed of finding solution in Δt with QoS. Considering these two ideas, an ABC based energy-efficiency live VM allocation policy with simulated annealing and gradient descent is proposed in this paper.

The main contributions of the paper are summarized as follows:

- (i) The idea of gradient descent is used to solve the problem that is how to satisfy the requirement of accelerating the speed of finding local optimum solution in a data center.
- (ii) The idea of simulated annealing is adopted to get m approximate global optimums with searching in turn per Δt duration.
- (iii) After finding global optimums in time, the frequency of VM migrations is reduced greatly to achieve a fairly high energy-efficiency level.

The rest of this paper is organized as follows. Related work is depicted in Section 2. The proposed VM allocation policy is given in Section 3. Simulated experiments are done in Section 4. Discussion and analysis are given in Section 5. Section 6 concludes the paper.

2. Related Work

The virtual machine allocation (VM allocation) is a process that is finding the target host for the live migrant VM. Traditionally, VM allocation policy is designed to achieve high performance. Currently, energy consumption is considered as another important influencing factor to evaluate the reasonability of VM allocation policy. Therefore, performance and energy consumption are two major influencing factors to construct energy-efficiency evaluation model.

Fan et al. [7] found a strong correlation between single server and CPU utilization rate; that is, there is a direct linear correlation between CPU utilization rate and the server's energy consumption. The higher the CPU utilization rate is, the more the energy consumption is. This is represented by a host energy evaluation model:

$$P_{\text{total}} = P_{\text{idle}} + (P_{\text{busy}} - P_{\text{idle}}) \mu_{\text{cpu}}, \quad (1)$$

where P_{total} represents the total energy consumption of a host, P_{idle} represents the energy consumption of an idle host, P_{busy} represents the energy consumption of a high load host, and

μ_{cpu} indicates the CPU utilization rate of a host. But it cannot measure the nonlinear CPU utilization rate.

Boglazov [8] proposed another empirical nonlinear model to evaluate the energy consumption of a host:

$$P_{\text{total}} = P_{\text{idle}} + (P_{\text{busy}} - P_{\text{idle}}) (2\mu_{\text{cpu}} - \mu_{\text{cpu}}^\gamma), \quad (2)$$

where γ represents calibration parameters for minimizing the squared error, which is obtained by experiments.

For the cost of energy consumption from data centers, researchers have found a variety of efficient policies to reduce the cost of energy and keep a relatively high QoS level.

Deng et al. [9] proposed the concept of green data centers, which means low energy consumption, low cost, and low levels of environment pollution. Fiandrino et al. [10] proposed the model of energy consumption for VM migration, which can reduce the cost of energy consumption of data centers to achieve workload balance. However, they did not discuss the problem of finding global optimum and unsuitable VM migration. Nakada and Hirofuchi [11] optimized the number of physical hosts and reduced the frequency of virtual machine migrations. They made Service Level Agreement (SLA) as an important influencing factor to measure the optimal results and also placed VM migration as a multiobjective optimization issue.

Nishant et al. [12] utilized the ant colony algorithm to solve the problem of VM migration. Each ant only updates its local data, which leads to the low rate of convergence. Jeyarani et al. [13] presented SAPSO to realize dynamic VM scheduling in data centers. SAPSO promptly detects and efficiently tracks the changing optimum that represents target servers for VM placement. However, the idea did not take future workload into account. From the perspective of long operation, the policy may incur additional costs.

Previously, our research focuses on making use of ABC algorithm to solve the problem of VM allocation, but it is insufficient on the speed of convergence for finding m local optimums in data centers. When dealing with the same volume of jobs, the higher the energy-efficiency level of target host is, the less it costs. As we all know, the energy-efficiency level will be decreased rapidly when the workload of a target host is larger than a certain critical value. Therefore, it is important to achieve a fairly high energy-efficiency level in data centers by making an effective live VM allocation policy. In our previous researches, ABC based energy-efficiency live VM allocation policy has been proposed. However, there are still two problems which need to be solved in Section 1.

3. ABC Based Live VM Allocation Policy with Simulated Annealing and Gradient Descent

Workload balance in data centers can save energy by migrating VMs to target hosts, and the key is to assign VM to the optimal target host [14]. To achieve the goal of saving energy in data centers, we introduce an ABC algorithm to seek the best target hosts. However, Bi and Wang [15] found that an ABC algorithm is easy to fall into local optimum. To solve this problem, simulated annealing makes the scouts jump out

of local optimal solution space and then optimize the optimal solution within multiple iterations. The following paper will elaborate it from two aspects, one is the energy evaluation model of data centers, and the other is an enhanced energy-efficiency live VM allocation policy.

3.1. Data Center Energy Evaluation Model. In the live VM allocation process, we want to assess the cost that the VM needs to be migrated to the target host. We use P_{diff} , which represents difference in the energy consumption, to evaluate virtual machine migration process cost:

$$\begin{aligned} P_{\text{after}} &= P_{\text{idle}} + (P_{\text{busy}} - P_{\text{idle}}) (2\mu_{\text{cpu.after}} - \mu_{\text{cpu.after}}^y) \\ P_{\text{now}} &= P_{\text{idle}} + (P_{\text{busy}} - P_{\text{idle}}) (2\mu_{\text{cpu.now}} - \mu_{\text{cpu.now}}^y) \\ P_{\text{diff}} &= 2(P_{\text{busy}} - P_{\text{idle}}) (\mu_{\text{cpu.after}} - \mu_{\text{cpu.now}}) \\ &\quad + (P_{\text{busy}} - P_{\text{idle}}) (\mu_{\text{cpu.now}}^y - \mu_{\text{cpu.after}}^y), \end{aligned} \quad (3)$$

where P_{after} corresponds to the energy consumption of a target host after receiving VM and P_{now} means the current energy consumption of the target host.

$$\varphi = \begin{cases} 1 & \text{if VM has been allocated to } H_j \text{ and } rcrVM \leq acrH_j \\ 0 & \text{if VM has been allocated to } H_j \text{ and } rcrVM \geq acrH_j \\ \text{Invalid} & \text{if VM has not been allocated to } H_j, \end{cases} \quad (5)$$

$$Z_s = \sum_{i=1}^n \sum_{j=1}^m \varphi, \quad (6)$$

$$\Delta E = \delta_k^r - \delta_{k-1}^r, \quad (7)$$

$$\delta E = \sum_{k=1}^m (\delta_k^r - \delta_{k-1}^r), \quad (8)$$

where $rcrVM$ represents the smallest computing nodes of VMs and $acrH_j$ marks the available computing resources of j th host. There are o positions for $\max \varphi$ that can be selected in S placement policy. It can be expressed as $PS = \{m, VM, o, t\}$.

The second parameter is based on energy consumption; $A = \{m, VM, r, t + t_k\}$, where r corresponds to any placement of o and k represents the host number. The position o , when selecting the host number k , is represented as $A = \{m, VM, r, t + t_k\}$. Their energy is represented as δ_k^y , and it means, according to r , the total energy consumption that is migrating VMs to the host. Hence migrating VM energy

consumption can be expressed as (7). In order to achieve better energy-efficiency, energy-efficiency VM migration can be represented as (8). In other words, in order to minimize energy consumption, we must minimize the value of δE .

On the basis of the energy evaluation model, the metric of fitness is defined as in

$$F_{\sigma, \omega} = \tan \theta = \frac{E_{\sigma} - E_{\omega}}{\sqrt{(x_{\sigma} - x_{\omega})^2 + (y_{\sigma} - y_{\omega})^2}}, \quad (4)$$

where E corresponds to the energy consumption and σ and ω mean two different physical hosts.

3.2. Formal Problem. We formalized the problem as migrating n virtual machines to m hosts. Its solution is represented as n -dimensional vector, where each element represents a target host. We can describe the problem as to find a virtual machine allocation policy; we define it as A , which is to satisfy the demand to maximize computing performance while minimizing energy consumption. To solve this problem, we define an array of $S = \{H, VM, E, A\}$. H represents m physical hosts; $H(m, t) = \{H_1, H_2, H_3, \dots, H_m\}$, where t means the VM migration time and VM represents n VMs. $VM(n, t, \Delta t) = \{VM_1, VM_2, \dots, VM_n\}$ is expressed as n VM migration demand in Δt . $E(m, t) = \{E_1, E_2, E_3, \dots, E_m\}$ represents the total energy consumption of m hosts. The purpose of the algorithm is to find the host location for maximum performance and minimize energy consumption. There are multiple locations to meet the demand, so we define as (6), $i \in \{1, 2, 3, \dots, n\}$, $j \in \{1, 2, 3, \dots, m\}$, $r \in \{1, 2, 3, \dots, s\}$, φ represents a different location selected that is migrating VM to number j host, and it has the following relationship:

consumption can be expressed as (7). In order to achieve better energy-efficiency, energy-efficiency VM migration can be represented as (8). In other words, in order to minimize energy consumption, we must minimize the value of δE .

3.3. Expressed Solutions. In order to achieve an ABC based energy-efficiency algorithm, representation of the solution becomes a priority. Solutions expressed represent the correspondence between the optimal bee and problem spaces found by ABC based algorithm. In this paper, there are n VMs to be migrated to m target hosts, so this is an n -dimensional vector problem. Each dimension is an integer

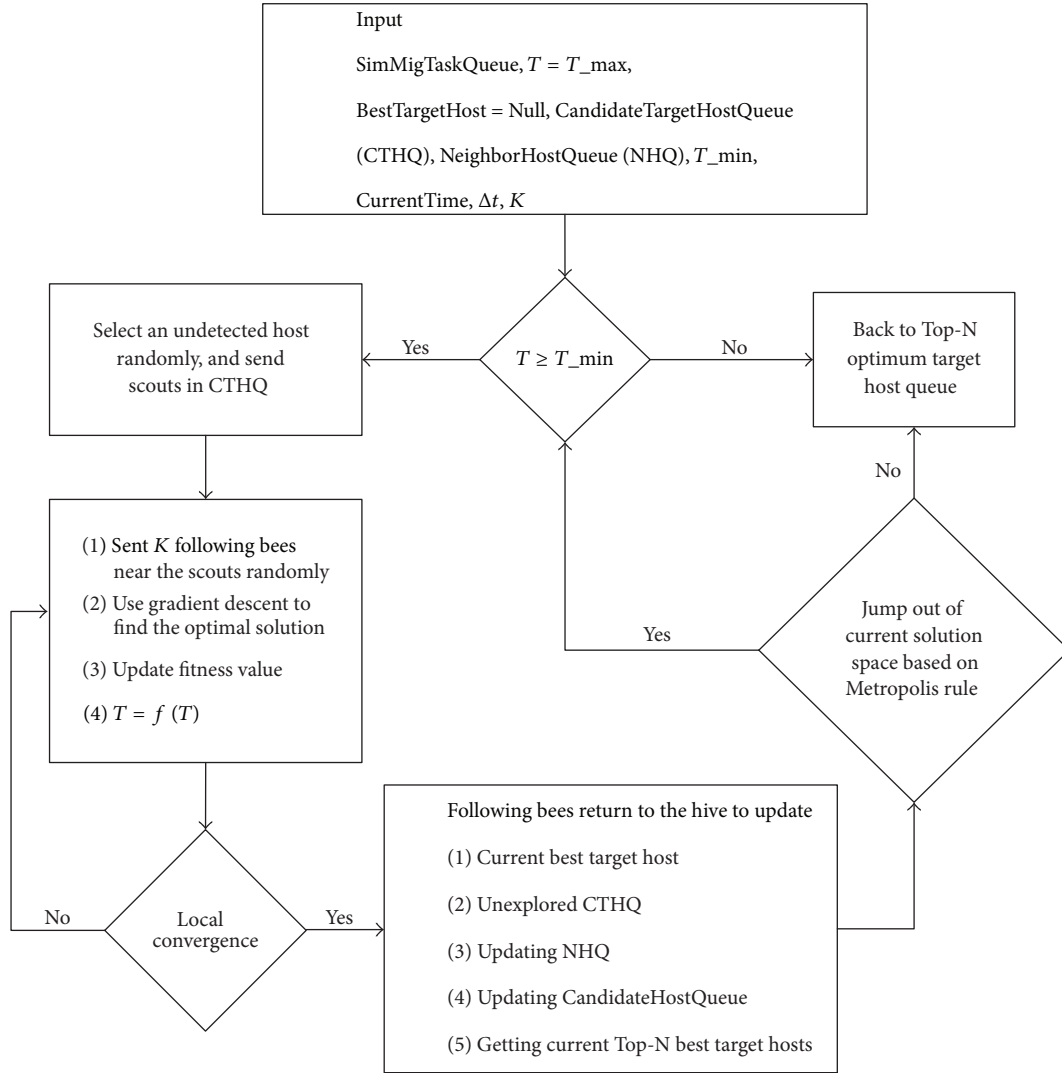


FIGURE 1: Flowchart of FP-ABC VM allocation policy.

from 0 to $m - 1$. Position vector is represented as $gBest_i = \{gB_{i1}, gB_{i2}, \dots, gB_{iD}\}$. v_{ij} shows a nectar of the i th (The i th solution vector) position of the j th (j th VMs to be migrated) dimension. Velocity vector is represented as $v_i = \{v_{i1}, v_{i2}, \dots, v_{ij}, \dots, v_{iD}\}$; v_{ij} shows j -dimensional speed of the i th nectar source. $pBest_i = \{pB_{i1}, pB_{i2}, \dots, pB_{iD}\}$ shows that the current optimal location is found, and best position vector is represented as $gBest_i = \{gB_{i1}, gB_{i2}, \dots, gB_{iD}\}$, which is regarded as the current best target location in the population.

3.4. The Main Idea of ABC Based Energy-Efficiency Live VM Allocation Policy. An ABC based live VM allocation policy has been considered as a better energy-efficiency policy. According to [15], it can reduce 25%~30% energy cost with acceptable SLA violation. However, in big data time, the ABC based energy-efficiency live VM allocation policy will face huge amount of data-intensive jobs and need to get the best VM allocation decision in time.

To solve these problems, we have improved the ABC algorithm in two ways. First, the traditional speed of ABC based algorithm uses fitness function to find local optima which need to be optimized. Therefore, the idea of gradient descent is applied to getting the local optima rapidly. Second, the traditional ABC based algorithm maybe falls into the local optima in a time window of Δt . Therefore, simulated annealing is applied to finding the global optima.

3.5. Improved Artificial Bees Live VM Allocation Policy. As illustrated in Figure 1, the FP-ABC VM allocation policy can be described as shown in Algorithm 1.

4. Results

4.1. Experimental Environment. The cloud simulator of CloudSim3.0 [3] is used to conduct a simulation experiment. Java programming language is adopted to get results from

Input: the candidate target host queue CTHQ = $\{h_1, h_2, \dots, h_n\}$
Output: *BestTargetHost*
Pseudo-code as follows:

- (1) CTHQ = $\{h_1, h_2, \dots, h_n\}$ are marked unexplored; Set the *SimMigTaskQueue*;
- (2) Set the maximum temperature (T_{\max}), $T = T_{\max}$ and T_{\min} ;
- (3) Initialize *CandidateTargetHostQueue*;
- (4) Set *NeighborHostQueue* to be *Null*, and set *BestTargetHost* also to be *Null*;
- (5) Set the number of following bees as K ;
- (6) Set the initial time point t , // set the number of n VM migration requests;
- (7) **While** ($T \geq T_{\min}$)
- (8) Select an undetected host randomly, and send scouts in candidate target host queue;
- (9) do
- (10) **Parbegin**
- (12) Calculates the host power // Sent K following bees near the scouts randomly
- (13) Use of gradient descent, calculation and evaluation function
- $$J(E) = \frac{1}{2} \sum_{i=1}^m (E(x_i, y_i))^2$$
- (14) **If** ($J(E) = 0$)
- (15) Output the host number of current nearby following bee // find current local optima
- (16) **else** {
- (17) in accordance with the rules of
- $$\min f \left(x_i + \alpha \frac{\partial}{\partial x_i} E(x_i, y_j) \right), \min f \left(y_j + \alpha \frac{\partial}{\partial x_i} E(x_i, y_j) \right)$$
- to apply the program of the gradient descent
- (18)
$$x_{i+1} = x_i - \alpha \frac{\partial}{\partial x_i} E(x_i, y_j), y_{j+1} = y_j - \alpha \frac{\partial}{\partial y_j} E(x_i, y_j)$$
- }
- (19) **Endif**
- (20) use
- $$E \left(x_i - \alpha \frac{\partial}{\partial x_i} E \right) = \min_{\alpha > 0} E \left(x_i - \alpha \frac{\partial}{\partial x_i} E \right)$$
 to update α_k ;
- (21) **ParEnd**
- (22) $T_k = T_0 / k^m$ // Make use of simulated annealing.
- (23) Update fitness value,
- $$F_{i,j} = \tan \theta = \frac{E_\sigma - E_\omega}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}};$$
- (24) **While** ($F_{i,j} = 0$)
- (25) Following bees have returned, update current best target host;
- (26) Update unexplored candidate target host group;
- (27) Update the nearby hosts, candidate host queue and *Top-N* current best target host queue
- (28) Recording time t ;
- (29) **Endwhile**
- (30) $\Delta E = E - E_0$ // Out of the current local optimal solution based on the Metropolis criterion
- (31) **If** ($\exp(-\Delta E/T) > \text{random}(0, 1)$)
- (32) Return to *Line* (7)
- (33) **else**
- (34) Output *BestTargetHost*
- (35) **Endwhile**
- (36) Output similar target hosts

ALGORITHM 1

MyEclipse9.0. The experiment is done on Lenovo™ desktop.
Its system parameters as follows:

CPU: Intel(R) Core(™) i7-3770 CPU @ 3.40 GHz.

RAM: 4.00 GB.

System type: 64-bit operating system.

OS: Windows 7.

PlanetLab [8] is a group of computers, which is tested for computer networking and distributed systems research. It was established in 2002 by Professor Larry L. Peterson and Professor David Culler, and it was composed of 1353 nodes at 717 sites worldwide in January 2016 [16]. 10-day sample data in PlanetLab cloud computing environment are chosen as experimental data as shown in Table 1 [8]. Simulation data center is

TABLE 1: Characteristics of the workload data (CPU utilization rate).

Date	The number of VMs	Mean	Standard deviation	Quartile 1	Median	Quartile 3
03/03/2011	1052	12.31%	17.09%	2%	6%	15%
06/03/2011	898	11.44%	16.83%	2%	5%	13%
09/03/2011	1061	10.70%	15.57%	2%	4%	13%
22/03/2011	1516	9.26%	12.78%	2%	5%	12%
25/03/2011	1078	10.56%	14.14%	2%	6%	14%
03/04/2011	1463	12.39%	16.55%	2%	6%	17%
09/04/2011	1358	11.12%	15.09%	2%	6%	15%
11/04/2011	1233	11.56%	15.07%	2%	6%	16%
12/04/2011	1054	11.54%	15.15%	2%	6%	16%
20/04/2011	1033	10.43%	15.21%	2%	4%	12%

TABLE 2: Power consumption by the selected servers at different workload levels in Watts.

Workload level	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
HP ProLiant G4	86	89.4	92.6	96	99.5	102	106	108	112	114	117
HP ProLiant G4	93.7	97	101	105	110	116	121	125	129	133	135

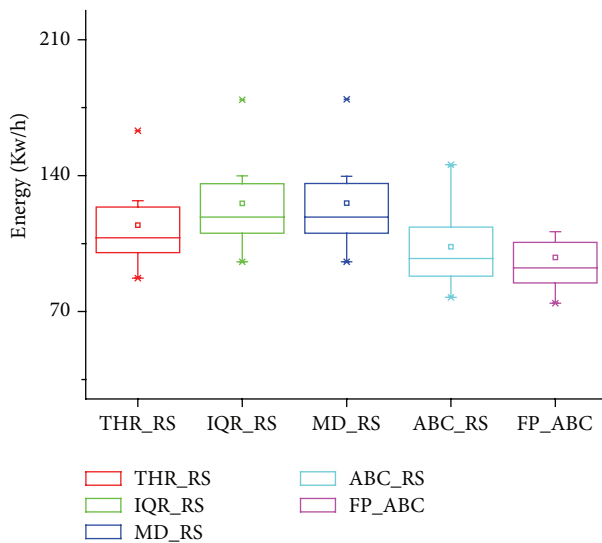


FIGURE 2: Energy consumption of the five algorithms.

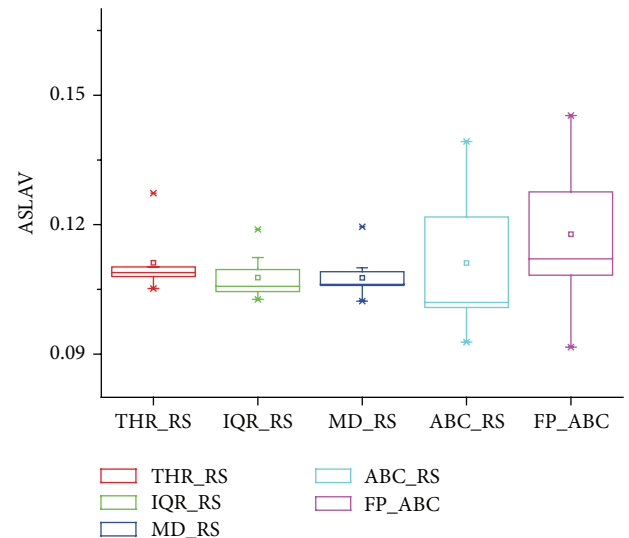


FIGURE 3: ASLAV metric of the five algorithms.

composed of 800 heterogeneous physical nodes, and half of these nodes are HP ProLiant ML100 G4 server (1860 MIPS) while the other halves are HP ProLiant ML100 G5 server (2660 MIPS) [8]. Energy consumption of these two servers at different loads is shown in Table 2. Server virtual machine is set as single-core; RAM can also be classified according to the number of VMs. The following are the main types of VMs: High-CPU Medium Instance (2500 MIPS, 0.85 GB), Extra Large Instance (2000 MIPS, 3.75 GB), Small Instance (1000 MIPS, 1.7 GB), and Micro Instance (500 MIPS, 613 MB).

The simulation is performed in CloudSim3.0, Δt is set as 30 seconds, and its safety parameter is 1.5.

4.2. Evaluating the Energy Consumption of FP-ABC with Different Approaches. In Figure 2, FP-ABC can reduce the

energy consumption of data center by 5.1% on the base of ABC_RS. The proposed energy evaluation model can be used to evaluate different VM allocation policies. Our proposed FP-ABC policy has advantage in saving energy.

4.3. Evaluating the Violation Rate of SLA with Different Approaches. Service Level Agreement (SLA) represents a guarantee under certain overload performance and reliability of service. In other words the service provider and the user-defined agreements are accepted by each other. SLAV is short for Violations of SLA. ASLAV is the average rate of SLAV. The QoS will be better if its ASLAV is lower. In Figure 3, FP-ABC is the last one. Among these traditional policies, THR_RS has the worst performance. And it proves that VM allocation policy of THR_RS gets higher ASLAV value. The

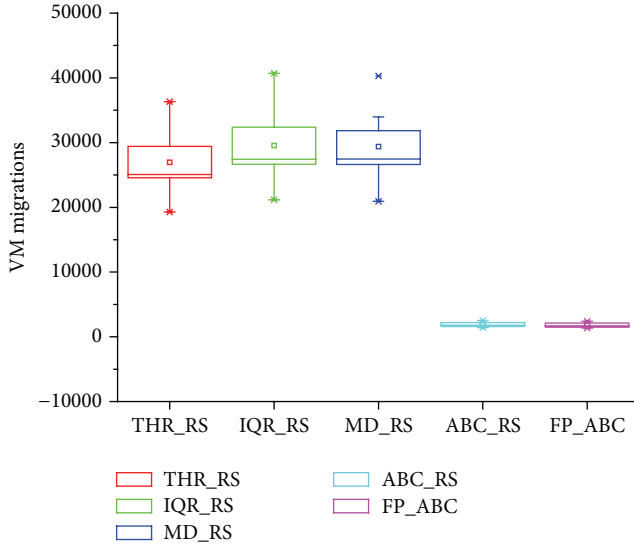


FIGURE 4: Number of VM migrations of the five algorithms.

ASLAV value of FP-ABC is 10.4%, and it means that FP-ABC can reduce energy consumption with acceptable ASLAV. Therefore, the workload balance based policy can satisfy user's requirement of QoS.

4.4. Evaluating the Effectiveness of FP-ABC with Different VM Allocation Policies. The number of VM migrations is a statistic variable to measure the VM migration times which occurred in data centers. The cost will be lower if fewer VM migrations occurred. As shown in Figure 4, FP-ABC and ABC_RS can reduce the frequency of VM migrations greatly. Furthermore, FP-ABC can reduce VM migrations by nearly 6.1% when compared with ABC_RS. An unreasonable VM allocation policy will increase the frequency of VM migrations. Therefore, FP-ABC policy can find best hosts as BestTargetHostList in Δt . As a result, we can save more energy.

4.5. Evaluating the Energy-Efficiency of FP-ABC with Different VM Allocation Policies. In formula (9), ESV is an energy-efficiency comprehensive index. It measures the tradeoff between performance and energy consumption. As we all know, the QoS will be violated when energy is saved greatly and vice versa. Therefore, achieving a good energy-efficiency tradeoff is difficult. However, FP-ABC has a good performance in the metric of ESV depicted in Figure 5. The first reason is that FP-ABC policy can reduce the VM migration frequency to keep live workload balance in data centers; the second reason is that the proposed energy consumption evaluation model can be used to make a better evaluation of energy consumption in data centers; the third reason is that idea of simulated annealing and gradient descent can improve the performance of ABC based policy. Evaluation model [8] is presented as follows:

$$ESV = EC * SLAV. \quad (9)$$

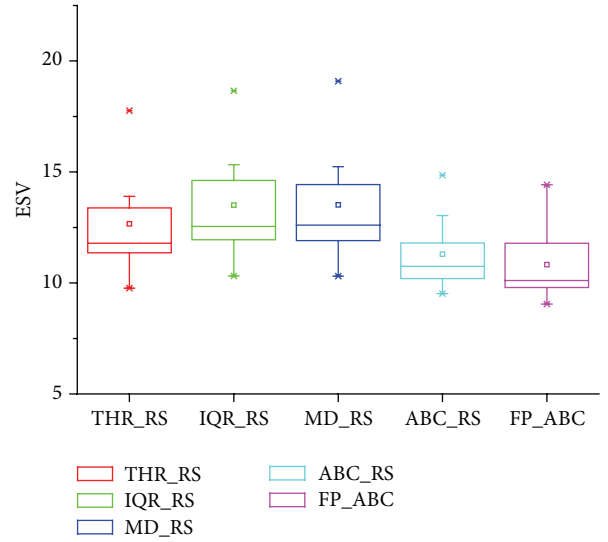


FIGURE 5: ESV metric of the five algorithms.

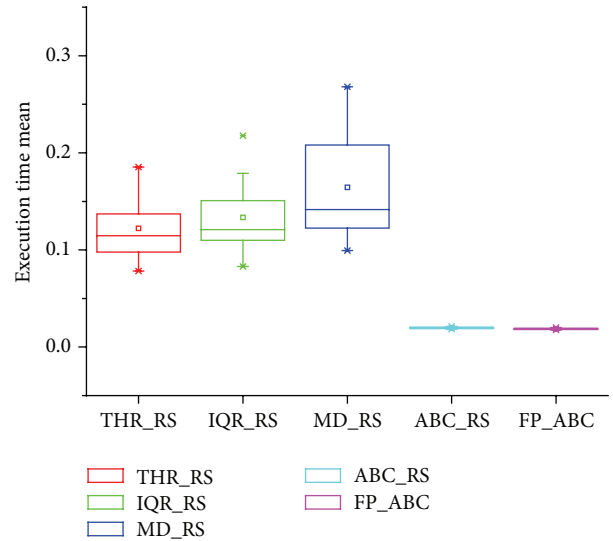


FIGURE 6: ETM metric of the five algorithms.

4.6. Evaluating the Decision Time of FP-ABC with Different VM Allocation Policies. In Figure 6, based on the index of execution time mean in Cloudsim3.0, it can reduce the decision time 85% to THR_RS, 86% to IQR_RS, 88% to MD_RS, and 13.8% to ABC_RS. Compared with these traditional migration policies, the FP-ABC is more effective to minimize the decision-making time.

5. Discussion

5.1. The Importance of Accelerating the Process of Finding Local Optimized Target Hosts with FP-ABC. A huge number of live migrant VMs are waiting to be allocated to their best target hosts in Δt . Therefore, the solutions should be given in some reasonable time window. Current state-of-the-art ABC based live VM allocation policy should be improved by accelerating

the speed of finding the local optimized target hosts. To solve this problem, the idea of gradient descent is adopted to ABC based live VM allocation policy. Current ABC based policy only makes the judgment based on its fitness value, without considering the requirement of speed. The idea of achieving a local solution quickly is realized by gradient descent. It will accelerate the process of searching local optimization in order to accelerate the speed of FP-ABC policy. In Section 4.6, we see that value of execution time mean is reduced when compared with 3 traditional policies and ABC_RS policy. It is significant when facing with amount of applications and also can save energy.

5.2. The Importance of Adopting Idea of Simulated Annealing to Find Global Optimized Target Hosts. We want to find the global optimum target host for each migrant VM in VM allocation, because the unreasonable target host will increase irrational energy consumption. In CloudSim3.0, three traditional VM migration policies are not good at finding the global optimum results, since their decision is made only by workload balance without reasonability analysis. Furthermore, the state-of-the-art ABC based policy is easy to fall into local optimum. Therefore, we add the idea of simulated annealing into ABC based policy, which can make scout bees jump out of the current solution space on a certain probability to detect other solution space.

5.3. Tradeoff between Energy Consumption and Service Quality. In data centers, to reduce energy consumption or improve the quality of service simply is not a good energy-saving policy, because reducing the total energy consumption will decrease service quality; that is to say, it will increase the value of SLAV. The main objective of this paper is to achieve energy efficient goal and make the value of SLAV accepted. Experimental results show that FP-ABC can save energy effectively in the balance of the EC and SLAV, which is based on formula (9) introduced in Section 4.5.

6. Conclusions

This paper presents a new live VM allocation policy to reach as far as possible to meet the SLAV value while saving energy consumption. We proposed the state-of-the-art ABC based policy with the idea of simulated annealing and the idea of gradient descent. The idea of gradient descent is used to find the optimal solution with higher speed. And the idea of simulated annealing is used to find the global optimal solutions in Δt . Simulation experiment has proved that FP-ABC policy can effectively reduce energy consumption with relative high level of QoS. In addition, FP-ABC can effectively reduce the energy consumption of data centers by 5.1%, when compared with the state-of-the-art algorithm of ABC_RS.

In the future, the proposed the state-of-the-art ABC based policy should be adapted to the cloud of cloud. Cloud of cloud is a new business model to get an economic data center with good service performance. The improved ABC based optimized model should be implemented in new environment.

Competing Interests

The authors declare that they have no competing interests.

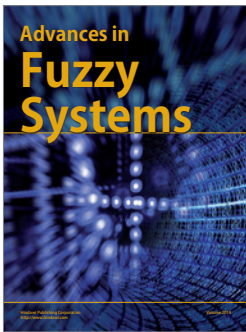
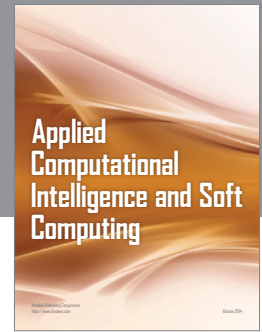
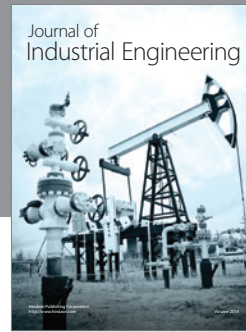
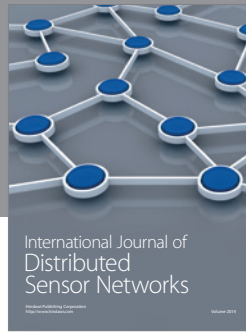
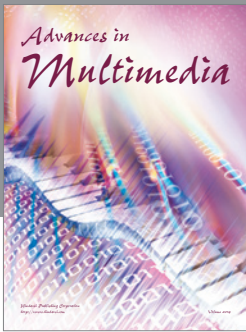
Acknowledgments

The authors are grateful to the financial support by the National Natural Science Foundation of China (nos. 61202306, 61472049, 61402193, and 61572225), the Foundation of Jilin University of Finance and Economics (nos. XJ2012007 and 201401), the Education Department of Jilin Province “Twelfth Five-Year” science and technology research projects under Grant no. 2015410, and Social Science Foundation of Jilin Province under Grant no. 2015BS48.

References

- [1] G. Semeraro, G. Magklis, R. Balasubramonian, D. H. Albonese, S. Dwarkadas, and M. L. Scott, “Energy-efficient processor design using multiple clock domains with dynamic voltage and frequency scaling,” in *Proceedings of the 8th International Symposium on High-Performance Computer Architecture (HPCA '02)*, pp. 29–40, Cambridge, Mass, USA, February 2002.
- [2] P. Barham, B. Dragovic, K. Fraser et al., “Xen and the art of virtualization,” in *Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP '03)*, pp. 164–177, 2003.
- [3] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, “CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms,” *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011.
- [4] J. Zhao, L. Hu, Y. Ding, G. Xu, and M. Hu, “A heuristic placement selection of live virtual machine migration for energy-saving in cloud computing environment,” *PLoS ONE*, vol. 9, no. 9, Article ID e108275, 2014.
- [5] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” Tech. Rep. TR06, Erciyes University, 2005.
- [6] J. Jiang, Y. Liu, L. Wang, J. Chen, N. Huang, and X. Wei, “ABCC: an energy-efficiency VM consolidation algorithm based on heuristic backward artificial bee colony method in data centers,” *Journal of Jilin University (Science Edition)*, vol. 6, pp. 1239–1248, 2014.
- [7] X. Fan, W. D. Weber, and L. A. Barroso, “Power provisioning for a warehouse-sized computer,” *ACM SIGARCH Computer Architecture News*, vol. 35, no. 2, pp. 13–23, 2007.
- [8] A. Boglazov, *Energy-Efficient Management of Virtual Machines in Data Centers for Cloud Computing*, The University of Melbourne, Victoria, Australia, 2013.
- [9] W. Deng, F.-M. Liu, H. Jin, and D. Li, “Leveraging renewable energy in cloud computing datacenters: state of the art and future research,” *Chinese Journal of Computers*, vol. 36, no. 3, pp. 582–598, 2013.
- [10] C. Fiandrino, D. Kliazovich, P. Bouvry, and A. Zomaya, “Performance and energy efficiency metrics for communication systems of cloud computing data centers,” *IEEE Transactions on Cloud Computing*, 2015.
- [11] H. Nakada and T. Hirofuchi, “Eliminating datacenter idle power with dynamin and intelligent VM relocation,” in *Distributed Computing and Artificial Intelligence*, vol. 79, pp. 645–648, Springer, Berlin, Germany, 2010.

- [12] K. Nishant, P. Sharma, V. Krishna et al., “Load balancing of nodes in cloud using ant colony optimization,” in *Proceedings of the 14th International Conference on Modelling and Simulation (UKSim '12)*, pp. 3–8, IEEE, Cambridge, UK, March 2012.
- [13] R. Jeyarani, N. Nagaveni, and R. Vasanth Ram, “Self adaptive particle swarm optimization for efficient virtual machine provisioning in cloud,” *International Journal of Intelligent Information Technologies*, vol. 7, no. 2, pp. 25–44, 2011.
- [14] R. Kapur, “A workload balanced approach for resource scheduling in cloud computing,” in *Proceedings of the 8th International Conference on Contemporary Computing (IC3 '15)*, pp. 36–41, IEEE, Noida, India, August 2015.
- [15] X. J. Bi and Y. J. Wang, “A modified artificial bee colony algorithm and its application,” *Journal of Harbin Engineering University*, vol. 33, no. 1, pp. 117–123, 2012.
- [16] PlanetLab, <http://planet-lab.org/>.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

