Correlation-Aware Virtual Machine Placement in Data Center Networks

Tao Chen[1], Yaoming Zhu[1], Xiaofeng Gao[1], Linghe Kong[1], Guihai Chen[1], and Yongjian Wang[2](✉)

[1] Shanghai Key Laboratory of Scalable Computing and Systems,
Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai 200240, China
{tchen,grapes_islet,linghe.kong}@sjtu.edu.cn,
{gao-xf,gchen}@cs.sjtu.edu.cn

[2] Key Laboratory of Information Network Security of Ministry of Public Security,
The Third Research Institute of Ministry of Public Security, Shanghai, China
wangyongjian@stars.org.cn

Abstract. The resource utilization (CPU, memory) is a key performance metric in data center networks. The goal of the cloud platform supported by data center networks is achieving high average resource utilization while guaranteeing the quality of cloud services. Previous work focus on increasing the time-average resource utilization and decreasing the overload ratio of servers by designing various efficient virtual machine placement schemes. Unfortunately, most of virtual machine placement schemes did not involve the service level agreements and statistical methods. In this paper, we propose a correlation-aware virtual machine placement scheme that effectively places virtual machines on physical machines. First, we employ Neural Networks model to forecast the resource utilization trend according to the historical resource utilization data. Second, we design correlation-aware placement algorithms to enhance resource utilization while meeting the user-defined service level agreements. The results show that the efficiency of our virtual machine placement algorithms outperform the previous work by about 15%.

Keywords: Virtual machine · Prediction · Correlation · Placement

1 Introduction

As the rapid development of cloud technology, data center networks (DCNs), the essential backbone infrastructure of cloud services such as cloud computing, cloud storage, and cloud platforms, attract increasing attentions in both academia and industry. Cloud data centers attempts to offer an integrated platform with a pay-as-you-go business model to benefit tenants at the same time, which is gradually adopted by the mainstream IT companies, such as Amazon EC2, Google Cloud Platform and Microsoft Azure. The multi-tenant and on-demand cloud service platform is achieved through virtualization on all shared resources

J. Wan et al. (Eds.): CloudComp 2016, SPNCE 2016, LNICST 197, pp. 22–32, 2018.
https://doi.org/10.1007/978-3-319-69605-8_3

and utilities, such as CPU, memory, I/O and bandwidth, in which various tenants buy virtual machines (VMs) within a certain period of time to run their applications [2]. Owing to multi-tenant demands, all kinds of workloads physically coexist but are logically isolated in DCNs, including data-intensive and latency-sensitive services, search engines, business processing, social-media networking, and big-data analytics. Elastic and dynamic resource provisioning is the basis of DCN performance, which is achieved by virtualization technique to reduce the cost of leased resources and to maximize resource utilization in cloud platforms. Therefore, the effectiveness of virtualization becomes essential to DCN performance.

Originally, the design goal of a DCN is to meet the peak workloads of tenants. However, at most time, DCNs are suffering from high energy cost due to low server utilization. A lot of servers are running with low workloads while consuming almost the same amount of energy as servers with high workloads. The cloud service providers have to spend more money on cooling bills to keep the servers in normal running. They aim to allocate resources in an energy-effective way while guaranteeing the Service Level Agreements (SLAs) for tenants.

A lot of literatures focus on enhancing the average utilization without violating SLAs. Some researchers focus on fair allocation schemes. Bobroff et al. [3] proposed a dynamic VM placement system for managing service level agreement (SLA) violations, which forecasts the future demand and models the prediction error. However, their approach only deals with single VM prediction, does not take correlation into consideration. Meng et al. [12] argued that VM should not be done on VM-by-VM basis and advocated joint-VM-provisioning, which can achieve 45% improvements in terms of overall utilization.

In this paper, we propose a correlation-aware virtual machine placement scheme that effectively places virtual machines on physical machines. First, we employ Neural Networks model to forecast the resource utilization trend according to the historical resource utilization data. Second, we design correlation-aware placement algorithms to enhance resource utilization while meeting the user-defined service level agreements. The simulation results show that the efficiency of our virtual machine placement scheme outperforms the previous work by about 15%.

The rest of the paper is organized as follows. Section 2 introduces the related work about resource demand prediction and virtual machine placement. Section 3 proposes the correlation-aware virtual machine placement system. Section 4 concludes this paper.

2 Related Work

2.1 Resource Demand Prediction

By appropriate prediction schemes, it is probable to mitigate hot spots in DCNs. Demand prediction methods will provide us early warnings of hot spots. Hence, we can adopt measures to ease the congestions in DCNs and allocate resource in a way that guarantee the performance of applications for tenants. The demand prediction methods usually fall into time series and stochastic process analyses.

The ARIMA model is often used to predict time series data. [3] forecasts the future demand and models the prediction error. However, their approach only deals with single VM prediction, does not take correlations between VMs into consideration. [11] accurately predicts the future VM workloads by seasonal ARIMA models. [13] employs SARMA model on Google Cluster workload data to predict future demand consumption. [14] uses a variant of the exponentially weighted moving average (EWMA) load predictor. For workloads with repeating patterns, PRESS derives a signature for the pattern of historic resource utilization, and uses that signature in its prediction. PRESS uses a discrete-time Markov chain with a finite number of states to build a short-term prediction of future metric values for workloads without repeating pattern, such as CPU utilization or memory utilization [7]. In [8], Markov chain model is applied to capture the temporal correlation of VM resource demands approximately.

2.2 Virtual Machine Placement

Virtual Machine Placement (VMP) is a problem involving mapping virtual machines (VMs) to physical machines (PMs). A proper mapping scheme can result in less PMs required and less energy cost. A poor resource allocation scheme may require more PMs and may induce more service level agreement (SLA) violations. Bobroff et al. [3] proposed a dynamic VM placement system for managing service level agreement (SLA) violations. They presented a method to identify servers which benefit most from dynamic migration. Meng et al. [12] argues that VM sizing should not be done on VM-by-VM basis and advocates joint-VM-provisioning which can achieve 45% improvements in terms of overall utilization. They first introduced a SLA model that map application performance requirements to resource demand requirement. Kim et al. [9] proposed a novel correlation-aware virtual machine allocation for energy-efficient datacenters. Specifically, they take correlation information of core utilization among virtual machines to consideration. Wang et al. [15] attempt to explore particle swarm optimization (PSO) to minimizing the energy consumption. They design an optimal VMP scheme with the lowest energy consumption. In [10], authors propose a VMP scheme which minimizes the energy consumption of the data center by consolidating VMs in a minimum number of PMs while respecting the latency requirement of VMs.

3 Correlation-Aware Virtual Machine Placement

3.1 System Architecture

We propose a correlation-aware virtual machine placement system for data center networks (DCNs) that predicts the future resource demand (utilization) of requests and minimize the number of physical machines (PMs) to meet the demand while considering the correlations between virtual machines (VMs) and satisfying a user-defined server level agreement (SLA) at the same time.

The system architecture is shown in Fig. 1, which includes three key components: monitor, predictor and controller. Tenants submit resource requests to the cloud platform. The cloud platform allocates the resources (VMs) for the requests. VMs are usually hosted on PMs in DCNs. Monitor module records the historical utilization data of VMs and transmit it to Predictor module. The predicted data generated from Predictor is delivered to Controller modular that makes a strategic decision for VM placement problem. An new VM placement strategy happens periodically every 100 time slots (a resource demand data recorded at a time slot).

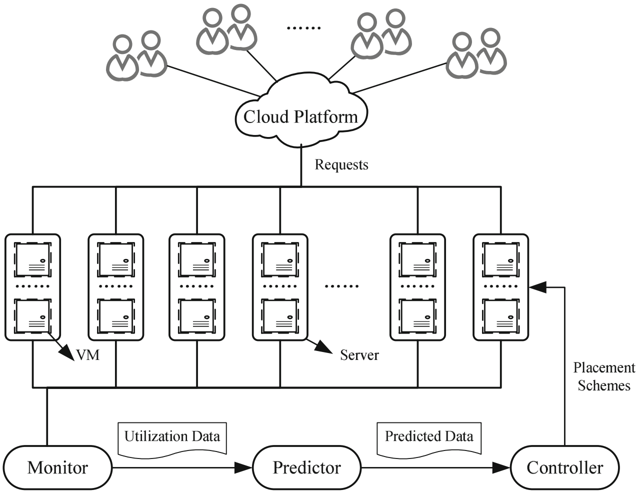

Fig. 1. Placement system architecture.

Traditionally, a VM placement scheme considers one VM at a time. In [12], the authors argued that the anti-correlation between VMs can be utilized. Their approach only picks two VMs at a time and allocate as less resource as possible for VMs. However, it is possible that three VMs that negatively correlate with each other, as shown in Fig. 2. Hence, we can do joint-provisioning of any number of VMs without SLA violations. The overall capacity allocated for VM 1, VM 2 and VM 3 under joint-provisioning is about 70% of a PM while the traditional VM placement needs to allocate about 85% capacity for these three VMs.

3.2 Prediction

In [16], the authors applied ARIMA and GARCH model to forecast the trend and volatility of the future demand. ARIMA performs well when an initial differencing step can be applied to remove non-stationarity. However, ARIMA is a

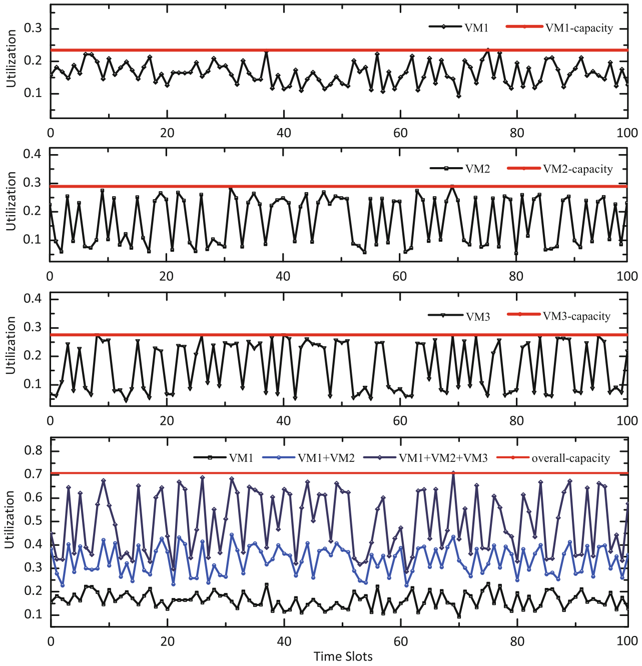

Fig. 2. VM correlation.

linear time series model and may not work otherwise. Neural Networks can be applied to predicted both linear and non-linear time series. For example, nonlinear autoregressive neural network (NARNET) can be trained to predict a time series from historical demand data.

Let NARNET(ni, nh) denotes a nonlinear autoregressive neural network with ni inputs and nh outputs. Such a model can be described as

$$U_i(t) = F(U_i(t-1), U_i(t-2), \ldots) + \varepsilon \tag{1}$$

where U_t is the variable of interest, and ε is the error term. We can the use this model to predict the value of U_{t+k}.

The performance of NARNET(10, 20) is shown in Fig. 3. The simulation results shows that NARNET can predict future resource demand accurately.

3.3 Virtual Machine Placement Algorithms

In this subsection, we present correlation-aware virtual machine placement algorithms. The allocated resource for VMs should match the future resource demand to achieve high resource utilization of PMs while meeting user-defined SLAs. Table 1 summarizes the main symbols used in this paper.

We use two performance metrics, overload ratio $\overline{o}$ and average resource demand $\overline{D}$, to evaluate the effectiveness of our proposed VM placement algorithms. The former is the ratio of the number of time slots when the actual

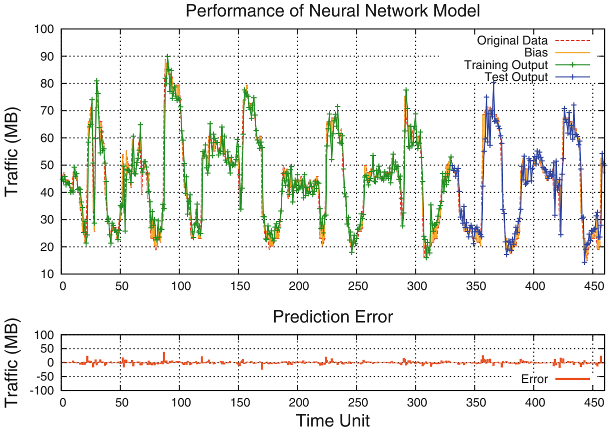

Fig. 3. Performance of NARNET.

Table 1. Main symbols and descriptions

Symbol	Description
$V = \{v_1, \cdots, v_n\}$	Set of VMs
$S = \{s_1, \cdots, s_m\}$	Set of PMs
N_{PM}	Number of used PMs for placement
D_m	Sum of resource demands in PM m
C	Capacity of a PM
$\overline{o}$	Overload ratio
$\overline{D}$	Average resource demand (utilization)
ϵ	User-defined SLA

resource demand of a PM is higher than its capacity over all the time slots × N_{PM}. The latter is the average resource utilization of PMs over all the time slots. The objective of algorithms is to achieve low overload ratio $\overline{o}$ and high average resource utilization $\overline{D}$. We monitor resource demand (e.g., CPU, memory) of each VM and predict conditional mean μ and the conditional variance σ. We also calculate the correlations ρ between different VMs placed on the same PMs according to resource demand time series data.

We can formulate the correlation-aware VM placement problem as follows.

$$\min N_{PM} \tag{2}$$

$$s.t. \quad \Pr(D_m > C) < \epsilon, \ \forall m, \tag{3}$$

$$\sum_m x_{mn} = 1, \ \forall n, \tag{4}$$

$$x_{mn} \in \{0, 1\}, \ \forall m, \ \forall n. \tag{5}$$

The binary variable x_{mn} indicates VM n is hosted on PM m or not. D_m denotes the resource demand of VMs on PM m. C means the capacity of a PM. $\epsilon > 0$ is a small constant, called *user-defined SLA*.

Equation (3) can be transformed to:

$$C \geqslant E[D_m] + c_\epsilon(0,1)\sqrt{var[D_m]}$$

$$E[D_m] = \mu_1 x_{m1} + \mu_2 x_{m2} + \ldots + \mu_n x_{mn},$$
$$var[D_m] = \sum_{i,j} \rho_{ij}\sigma_i\sigma_j x_{mi}x_{mj}.$$

where $c_\epsilon(0,1)$ is the $(1-\epsilon)$-percentile of standard normal distribution with mean 0 and variance 1. For example, when $\epsilon = 2\%$, $c_\epsilon(0,1) = 2.06$. $E[D_m]$ is the sum of expectations of resource demands of all VMs placed on PM m, and $var[D_m]$ is the variance of the workload with correlations between VMs taken into consideration.

After problem formulation, we will present our algorithms to the VM placement problem. The first algorithm is *Correlation-Aware First-Fit* algorithm. The algorithm is similar to first-fit algorithm in solving the bin-packing problem, which is shown in Algorithm 1.

Algorithm 1. Correlation-aware First-Fit VM Placement Algorithm

```
Input: Historical resource demand data of VMs from the monitor.
Output: A VM placement scheme with a user-defined SLA.
foreach VM $n$ do
    foreach PM $m$ do
        Add VM $n$ to PM $m$;
        $x_{mn} = 1$;
        if $E[D_m] + c_\epsilon(0,1)\sqrt{var[D_m]} < C$ then
            break;
        else
            Remove VM $n$ from PM $m$;
            $x_{mn} = 0$;
        end
    end
end
```

Algorithm 1 is a first-fit algorithm which will place a certain VM into the first PM that can hold it with a certain probability less than a user-defined SLA. Since this problem is very similar to first-fit algorithm of bin packing problem, we can easily reach the inequality the number of PMs used by first-fit described above is no more than 2× optimal number of PMs. If we first sort the VMs by the size, then this is very similar to first fit decreasing algorithm in bin packing problem. It has been shown to use no more than $\frac{11}{9}\mathbf{OPT} + 1$ bins (where $\mathbf{OPT}$ is the number of bins given by the optimal solution).

The second algorithm is *Correlation-Aware Best-Fit* algorithm, as shown in Algorithm 2. The main idea is: each packing is determined in a search procedure

Algorithm 2. Correlation-Aware Best-Fit VM Placement Algorithm

```
Input: Historical resource demand data of VMs from the monitor.
Output: A VM placement scheme with a user-defined SLA.
foreach VM n do
    Try to place VM n on every PM, and finally chose the PM m with the least slack to
      place;
    $x_{mn} = 1$;
    if $E[D_m] + c_\epsilon(0,1)\sqrt{var[D_m]} < C$ then
        break;
    else
        Remove VM n from PM m;
        $x_{mn} = 0$;
    end
end
```

that tests all possible subsets of items on the list which fit the bin capacity. We will choose the subset with the least slack to fill the bin. If the algorithm finds a subset that fills the bin completely, the search is stopped, for there is no better packing possible.

We compare our VM placement algorithms with the following benchmark algorithms:

Random. It is based on the idea of randomly place VMs to PMs according the peak value in historical resource demand data without making any predictions and considering correlations between VMs.

Constant variance (CV). This algorithm predicts the future demand of VMs while not taking correlations between VMs into consideration [16].

3.4 Evaluation

The resource demand (utilization) data is generated by the method in [1]. We put 384 VMs on 128 PMs. We first generate 200 resource demand traces with different mean and variation. Each trace contains a list of 400 historical resource demand data (400 time slots). We will use the first 100 data to train the neural network model and the remaining data to compare our correlation-aware placement algorithms with previous proposed algorithms. We normalize the capacity of a PM as 100%.

As shown in Table 2 and Fig. 4, when the user-defined SLA becomes larger, there are more PMs that achieve average resource utilization. There is a trade-off between resource utilization and SLA guarantee, and we should think twice before we make the decision under different scenarios.

As shown in Fig. 5, the resource utilizations of PMs are different under the four algorithms with user-defined SLA 5%. The random algorithm randomly place VMs to PMs according to the peak value in the historical data. Hence, the average resource utilization of PMs is the lowest among the four algorithm and the number of used PMs are the largest. The constant variance algorithm assumes the variance of VM i is constant which is apparently not the case in the real world. Correlation-aware first-fit and best-fit algorithms outperforms

Table 2. Number of used PMs, the overload ratio ($\overline{o}$), and average resource utilization of PMs ($\overline{D}$) in different time slots under different user-defined SLAs.

	Time Slots 100–200			Time Slots 200–300			Time Slots 300–400		
$\epsilon = 2\%$	N_{PM}	$\overline{o}$	$\overline{D}$	N_{PM}	$\overline{o}$	$\overline{D}$	N_{PM}	$\overline{o}$	$\overline{D}$
FF	95	0.76%	66.87%	96	0.69%	66.17%	96	0.69%	66.15%
BF	95	0.72%	66.87%	96	0.63%	66.17%	96	0.71%	66.15%
CV	128	0.007%	49.63%	128	0%	49.63%	128	0%	49.61%
$\epsilon = 5\%$	N_{PM}	$\overline{o}$	$\overline{D}$	N_{PM}	$\overline{o}$	$\overline{D}$	N_{PM}	$\overline{o}$	$\overline{D}$
FF	90	2.22%	70.58%	91	1.74%	69.81%	92	1.73%	69.03%
BF	90	2.27%	70.58%	91	1.84%	69.81%	92	1.72%	69.03%
CV	96	2.4%	66.17%	96	2.74%	66.17%	96	2.95%	66.15%
$\epsilon = 10\%$	N_{PM}	$\overline{o}$	$\overline{D}$	N_{PM}	$\overline{o}$	$\overline{D}$	N_{PM}	$\overline{o}$	$\overline{D}$
FF	85	6.36%	74.74%	87	4.9%	73.02%	86	5.32%	73.84%
BF	85	6.63%	74.74%	87	4.9%	73.02%	86	5.32%	73.84%
CV	96	2.65%	66.17%	96	2.65%	66.17%	96	2.95%	66.15%
Random	128	6.02%	52.5%	128	5.74%	52.37%	128	5.94%	52.48%

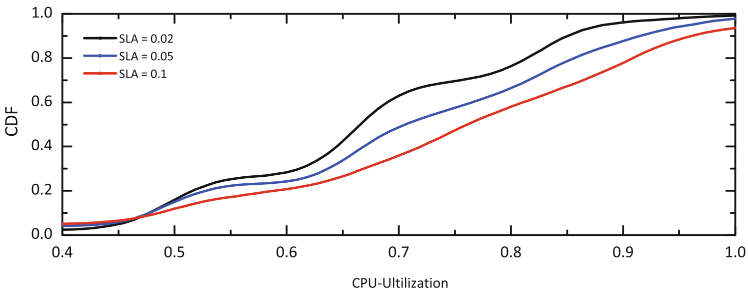

Fig. 4. Correlative-aware algorithms with different user-defined SLAs.

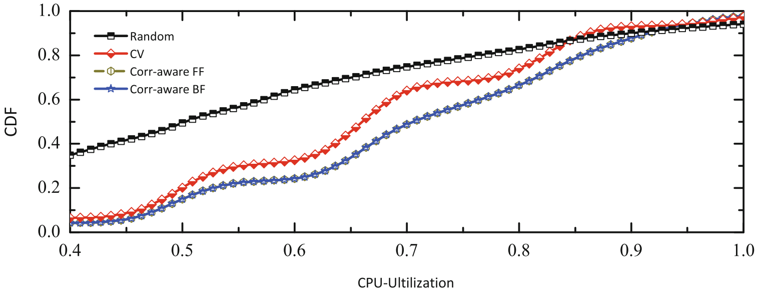

Fig. 5. Resource utilization with user-defined SLA 5%.

the other two algorithms. There are more PMs with high resource utilization and reduces the total number of used PMs. Tough the resource utilizations of PMs are almost the same, the best-fit algorithm costs more than the first-fit algorithm due to the test of placing a VM on every PM.

4 Conclusion

In this paper, we proposed a correlation-aware virtual machine placement system that effectively places virtual machines on physical machines. First, we employ Neural Networks model to predict the resource utilization trend according to the historical resource utilization data. Second, we presented two correlation-aware placement algorithms to enhance resource utilization while meeting the user-defined service level agreements. The simulation results show that the efficiency of our virtual machine placement scheme outperforms the previous work by about 15%.

Acknowledgement. This work has been supported in part by the China 973 Project (2014CB340303), China NSF Projects (Nos. 61672353, 61472252, 61672349 and 61303202), the Opening Project of Key Lab of Information Network Security of Ministry of Public Security (the Third Research Institute of Ministry of Public Security) (Grant no. C15602). The authors also would like to thank Wei Wei and Bo Cao for their contributions on the early versions of this paper.

References

1. Ajiro, Y., Tanaka, A.: Improving packing algorithms for server consolidation. In: International CMG Conference, vol. 253 (2007)
2. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Future Gener. Comput. Syst. **25**(6), 599–616 (2009). Elsevier
3. Bobroff, N., Kochut, A., Beaty, K.: Dynamic placement of virtual machines for managing SLA violations. In: IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 119–128. IEEE Press (2007)
4. Cao, B., Gao, X., Chen, G., Jin, Y.: NICE: network-aware VM consolidation scheme for energy conservation in data centers. In: IEEE International Conference on Parallel and Distributed Systems (ICPADS), pp. 166–173. IEEE Press (2014)
5. Clark, C., Fraser, K., Hand, S., Hansen, J.G., Jul, E., Limpach, C., Warfield, A.: Live migration of virtual machines. In: USENIX Proceedings of the 2nd Conference on Symposium on Networked Systems Design & Implementation (NSDI), pp. 273–286. USENIX Association (2005)
6. Ghorbani, S., Schlesinger, C., Monaco, M., Keller, E., Caesar, M., Rexford, J., Walker, D.: Transparent, live migration of a software-defined network. In: ACM Symposium on Cloud Computing (SOCC), pp. 1–14. ACM (2014)
7. Gong, Z., Gu, X., Wilkes, J.: Press: predictive elastic resource scaling for cloud systems. In: International Conference on Network and Service Management (CNSM), pp. 9–16. IEEE Press (2010)

8. Han, Z., Tan, H., Chen, G., Wang, R., Chen, Y., Lau, F.: Dynamic virtual machine management via approximate Markov decision process. In: IEEE International Conference on Computer Communications (INFOCOM), pp. 1–9. IEEE Press (2016)
9. Kim, J., Ruggiero, M., Atienza, D., Lederberger, M.: Correlation-aware virtual machine allocation for energy-efficient datacenters. In: Proceedings of the Conference on Design, Automation and Test in Europe, pp. 1345–1350. EDA Consortium (2013)
10. Khalilzad, N., Faragardi, H.R., Nolte, T.: Towards energy-aware placement of real-time virtual machines in a cloud data center. In: IEEE High Performance Computing and Communications (HPCC), pp. 1657–1662. IEEE Press (2015)
11. Lin, H., Qi, X., Yang, S., Midkiff, S.: Workload-driven VM consolidation in cloud data centers. In: Parallel and Distributed Processing Symposium (IPDPS), pp. 207–216. IEEE Press (2015)
12. Meng, X., Isci, C., Kephart, J., Zhang, L., Bouillet, E., Pendarakis, D.: Efficient resource provisioning in compute clouds via VM multiplexing. In: International Conference on Autonomic Computing, pp. 11–20. ACM (2010)
13. Qiu, C., Shen, H., Chen, L.: Probabilistic demand allocation for cloud service brokerage. In: IEEE International Conference on Computer Communications (INFOCOM), pp. 1–9. IEEE Press (2016)
14. Song, W., Xiao, Z., Chen, Q., Luo, H.: Adaptive resource provisioning for the cloud using online bin packing. IEEE Trans. Comput. **63**(11), 2647–2660 (2014). IEEE Press
15. Wang, S., Liu, Z., Zheng, Z., Sun, Q., Yang, F.: Particle swarm optimization for energy-aware virtual machine placement optimization in virtualized data centers. In: Parallel and Distributed Systems (ICPADS), pp. 102–109. IEEE Press (2013)
16. Wei, W., Wei, X., Chen, T., Gao, X., Chen, G.: Dynamic correlative VM placement for quality-assured cloud service. In: IEEE International Conference on Communications (ICC), pp. 2573–2577. IEEE Press (2013)
17. Wood, T., Shenoy, P., Venkataramani, A., Yousif, M.: Black-box and gray-box strategies for virtual machine migration. In Proceedings of the 4th USENIX conference on Networked systems design & implementation (NSDI) pp. 17–17. USENIX Association (2007)
18. Wood, T., Ramakrishnan, K.K., Shenoy, P., Van der Merwe, J., Hwang, J., Liu, G., Chaufournier, L.: CloudNet: dynamic pooling of cloud resources by live WAN migration of virtual machines. IEEE/ACM Trans. Netw. (TON) **23**(5), 1568–1583 (2015)
19. Xu, F., Liu, F., Liu, L., Jin, H., Li, B., Li, B.: iAware: making live migration of virtual machines interference-aware in the cloud. IEEE Trans. Comput. (TOC) **63**(12), 3012–3025 (2014)
20. Ye, K., Jiang, X., Huang, D., Chen, J., Wang, B.: Live migration of multiple virtual machines with resource reservation in cloud computing environments. In: IEEE International Conference on Cloud Computing (CLOUD), pp. 267–274. IEEE Press (2011)