PROCEEDINGS OF SCIENCE

An Energy-Efficient VM Consolidation Algorithm for Cloud Data Centers

Xiaodong Wu[1]
Faculty of Mathematics and Computer Science, Quanzhou Normal University
Fujian Provincial Key Laboratory of Data Intensive Computing
Quanzhou, 362000, China
E-mail: xdwu@foxmail.com

Cloud data centers consume vast amounts of electrical energy all over the world. Energy conservation in data centers has become a severe problem. This paper presents an energy-efficient virtual machine (VM) consolidation algorithm for cloud data centers. The proposed policy exploits the average utilization of each VM to estimate the expected utilization of physical hosts. Then, the expected utilization, the actual utilization and the static threshold are combined together to determine whether a host is at the status of overloading or not. Some VM(s) on the overloading hosts are then selected for migration to avoid Service Level Agreements (SLA) violations. The results of the experiment show that the proposed algorithm can effectively reduce energy consumption while guarantee SLAs.

ISCC2017
16-17 December 2017
Guangzhou, China

[1]Speaker

This work is supported in part by the National Natural Science Foundation of China under Grants (61472150), by the Natural Science Foundation of Fujian Province, China (2015J01663), by the Key Project of Quanzhou City Science and Technology Program (2015Z121, 2014Z134).

1.Introduction

In recent years, cloud computing has received significant attention. However, as large-scale virtualized data center consisted of tens of thousands of phisical hosts grows, enormous energy has been consumed for computing and equipment cooling. Energy conservation has become an critical issue.

Virtual machine (VM) consolidation has been shown to be one of the effective techniques for energy conservation [1]. There are many research works been done for energy-efficient VM scheduling problem. For example, Dabbagh et al. [2] have proposed an energy-aware VM migration and placement framework to reduce the number of active physical servers for overcommitted clouds. Xiong et al. [3] have put forward an energy-efficient VM allocation algorithm based on a multiresource allocation model and the particle swarm optimization (PSO) technique. Considering for both energy-efficient and SLA violation reduction, Zhou et al. [4] have presented adaptive three-threshold algorithms to determine thresholds. Beloglazov and Buyya [5, 6] have proposed static and double-threshold-based VM scheduling algorithms to improve energy efficiency and ensure SLAs in data centers. Cao and Dong [7] have proposed a energy-aware VM consolidation policy to utilize the mean and standard deviation of host utilizatioins to determine whether the host is overloaded.

This paper focus on the energy- and QoS- aware VM schuduling problem in cloud data centers. A threshold-based scheduling algorithm is proposed to exploit the average utilization of VMs to estimate the expected utilization of each host. Then, the expected utilization, the actual utilization and the threshold are combine together to determine whether a host is overloaded or not. The results of the experiment have shown that the proposed algorithm can effectively reduce the energy cnosumption while ensure SLAs.

2.Preliminary

2.1 System Model

Suppose there are *m* physical hosts in the data center: $\{P_1, P_2, \ldots, P_m\}$ and nv_i virtual machines are placed onto host P_i. The j-th virtual machine allocated on host P_i is denoted as $V_{i,j}$. The performance is mainly defined by MIPS, RAM and network bandwidth. The maximum CPU capacity of a virtual machine V is denoted by $vmips(V)$. The utilization of a virtual machine V at time point t is denoted by $u(V,t)$. Moreover, according to certain server consolidation policies, VMs can be migrated from one host to another on the basis of their dynamic utilizations, so as to reduce the number of active hosts to save energy.

Some symbols and their definitions used throughout the paper are summarized in Table 1.

2.2 Power Model

The power consumption of a host is primarily dependent on its CPU utilization. In this work, we adopt the real data on power consumption provided by the SPECpower benchmark [8,5]. In this work, two kinds of servers are selected, i.e., HP ProLiant ML110 G4 G5. The configurations of them are Intel Xeon 3040, 2 cores×1860 MHz, 4 GB, and Xeon 3075, 2 cores×2660 MHz, 4 GB, respectively.

The energy comsumption of all hosts within the datacenter during a time interval $[t_1, t_2]$ is calculated by

$$Energy=\int_{t_1}^{t_2}(\sum_{i=1}^{m} pow(p_i,t))\,dt \tag{2.1}$$

where m is the number of physical hosts and $pow(P_i,t)$ is the power consumption of host P_i at time t.

Symbol	Definition
m	The number of physical hosts in the data center
P_i	The i-th host
nv_i	The number of virtual machines allocated on host P_i
$V_{i,j}$	The j-th virtual machine on P_i
$u(V,t)$	The utilization of virtual machine V at time point t
$hmips(P_i)$	The CPU performance of P_i, defined in MIPS (Millions Instructions Per Second)
$vmips(V)$	The maximum CPU capacity of virtual machine V, defined in MIPS
$avgu(V)$	The mean utilization of VM V
$expu(P_i)$	The expected values for the utilizations of host P_i

Table 1: Symbols and Definitions

3. Virtual Machine Consolidation Algorithm

Virtual machine migration is one of the effective methods to improve the energy efficiency in data centers. Generally, this problem can be splitted into three steps: (1) determining whether a host is at the status of overloading or underloading, such that some or all VMs on this host should be migrated, respectively; (2) determining which VM(s) should be migrated; (3) selecting new hosts to accommodate the VMs selected for migration. The proposed THRAU (THReshold- and Average Utilization-based) scheduling policy mainly aims at step 1, that is, host overloading detection.

3.1 The THRAU Host Overloading Detection

In order to predict whether a host will be overloaded, the utilization of each VM V at n time points $\{t_{1,}t_{1,}..t_n\}$ are collected and used as samples. Suppose the set of utilizations of VM V at n time points is $\{u(V,t_1),u(V,t_2),\ldots u(V,t_n)\}$. Then, the mean utilization of V, $avgu(V)$, can be calculated by Equation (2).

$$avgu(V)=\frac{1}{n}\times\sum_{i=1}^{n}u(V,t_i) \tag{3.1}$$

As the mean utilizations of all VMs on host P_i is obtained, the expected value for the utilization of hosts P_i, $expu(P_i)$, is figured out by Equation (3).

$$expu(P_i)=\frac{\sum_{j=1}^{nv_i}(avgu(V_{i,j})\times vmips(V_{i,j}))}{hmips(P_i)} \tag{3.2}$$

Then, the expected utilizations of the hosts are combined with the (static) threshold and the actual utilizations to decide whether a host is at the state of overloading. The pseudo-code for the proposed overloading detection policy THRAU is presented in Algorithm 1.

Firstly, the expected utilization of the host is calculated from line 4 to line 8 by Equation (3). Line 9-13 of the Algorithm calculates the actual utilization of the host, which is the actual total required MIPS devided by the maximum CPU capacity . Next, let *maxu* and *minu* be the larger and the smaller one (line 14-15) of the expected and the actual utilizations. Then, the predicted utilization that is used to make prediction is figured out by:

$$predU = minu + (maxu - minu) \times threshold \quad .$$

Finally, *predU* is used as a upper threshold to decide the status of the host.

As can be seen from line 14 to line 18 of the pseudo-code, three factors, i.e., the expected utilization (line 8), the actual utilization (line 13) and the static threshold, are integrated by the algorithm to determine the status of the host.

Algorithm 1: Threshold and Average Utilization based overloading detection

```
1.  Input: host Pi, static threshold threshold and current time point t;
2.  Output: the status of the host Pi;
3.  Begin
4.    expmips = 0;
5.    Foreach virtual machine V on Pi do          // sum up the expected MIPS
6.      expmips += avgu(V)×vmips(V);
7.    End Foreach
8.    expUtilization = expmips ÷ hmips(Pi);       //expected utilization
9.    reqmips = 0;
10.   Foreach virtual machine V on Pi do          // sum up the actual requested MIPS
11.     reqmips += u(V,t) × vmips(V) ;
12.   End Foreach
13.   actUtilization = reqmips ÷ hmips(Pi);       //actual utilization
14.   maxu = Max (expUtilization, actUtilization);
15.   minu = Min (expUtilization, actUtilization);
16.   predUtilization = minu + (maxu – minu) × threshold;  //predicted utilization
17.   If (predUtilization > threshold ), then Pi is overloading
18.   Else Pi is not overloading.
19.   End If
20. End
```

Note that the THRAU policy detects the overloading hosts from the point of VMs rather than from the point of physical hosts. Hence, it can be considered as a relatively fine-grained algorithm. Consequently, the overloading prediction and detection can be more precisely identified.

3.2 The THRAU-MMT Scheduling Algorithm

Since the MMT policy (Minimum Migration Time) produces better results compared to other policies [5], MMT is utilized to select VM(s) to be migrated in this work. Moreover, the Power Aware Best Fit Decreasing (PABFD) algorithm [9] and the underloading detection policy [5] are respectively used for finding placement for migration VMs and detect underloading hosts. Hence, the proposed energy-efficient scheduling algorithm is called as THRAU-MMT in this paper.

3.3 Metrics

It is of great importance to meet QoS (Quality of Service) requirements in cloud environments. QoS requirements can be defined in term of Service Level Agreements (SLAs) . Therfore, SLA violation is an important factor for VM consolidation algorithms. In this work, several metrics with regards to QoS and SLA violation are defined to measure SLAs.

SLATAH (SLA violation Time per Active Host) is defined as the mean of the percentage of total SLA violation time of all hosts [5]:

$$SLATAH = \frac{1}{m} \sum_{i=1}^{m} \frac{T_{s_i}}{T_{a_i}} \tag{3.3}$$

where T_{s_i} represents the total time when P_i has experienced the utilization of 100% such that a SLA violation occurs and T_{a_i} is the total time when P_i is in the active state.

PDM (Performance Degradation due to Migrations) is defined as overall performance degradation by VM migrations [5]:

$$PDM = \frac{1}{\sum_{i=1}^{m} nv_i} \sum_{i=1}^{m} \sum_{j=1}^{nv_i} \frac{d_{i,j}}{r_{i,j}} \tag{3.4}$$

where nv_i is the number of VMs on host P_i , $d_{i,j}$ is the estimate of the performance degradation of $V_{i,j}$, and $r_{i,j}$ the total CPU capacity requested by $V_{i,j}$.

Then, the SLA violation is defined as:

$$SLAV = SLATAH \times PDM \tag{3.5}$$

Finally, the metric that includes both energy consumption and SLA violation is:

$$ESV = SLAV \times Energy \tag{3.6}$$

where *Energy* can be obtained by Equation (1).

4. Performance Evaluation

In order to envaluate the performance of the proposed algorithm, we have used a popular simulator, CloudSim toolkit [10], to conduct experiments. The performance of the traditional DVFS approach (without server consolidation), the THR-MMT [5] and the proposed THRAU-MMT algorithms are evaluated in the experiments.

The simulated data center includes 800 physical servers. Half of the servers are HP ProLiant G4 and the others are HP ProLiant G5. The workload used in the experiments is the data of 03/25/2011, coming from the CoMon project which is a monitoring infrastructure for PlanetLab [11]. Moreover, the $avgu(V)$ of each VM is calculated based on the utilization values gathered every 5 minutes in 24 hours.

Table 2 summarizes the performance comparison of DVFS, THR-MMT and THRAU-MMT algorithms with different thersholds. DVFS consumes the most energy since the VM consolidation policy has not been applied. However, DVFS generates no SLA violation. This is because no VM migration strategy is used according to DVFS.

Threshold	Energy (kWh)			SLAV ($\times10^{-4}$)		PDM ($\times10^{-3}$)		ESV ($\times10^{-2}$)	
	DVFS	THR-MMT	THRAU-MMT	THR-MMT	THRAU-MMT	THR-MMT	THRAU-MMT	THR-MMT	THRAU-MMT
0.5	785.49	226.437	212.436	0.654	0.578	1.15	1.04	1.482	1.228
0.6	785.49	201.904	192.839	0.522	0.457	0.93	0.85	1.053	0.880
0.7	785.49	188.670	179.296	0.387	0.401	0.74	0.77	0.731	0.719
0.8	785.49	175.431	168.355	0.346	0.358	0.68	0.70	0.607	0.602
0.9	785.49	163.702	157.338	0.331	0.508	0.65	0.89	0.542	0.800
1.0	785.49	152.803	137.345	2.503	1.516	1.75	1.50	3.824	2.082

Table 2: Performance Comparison of the Two VM Scheduling Algorithm

Moreover, as can be seen from Figure 1(b-d), considering the SLAV, PDM and ESV metrics related to QoS, THRAU-MMT obtains equal or better overall performance than THR-MMT algorithm. However, as shown in Figure 1(a), compared with THR-MMT, THRAU-MMT consumes less energy. Up to 10.1% energy consumption can be saved by THRAU-MMT. The reason is that the THRAU policy analyzes and predicts the host status in a fine-grained manner, such that the host status prediction can be more accurately. As a result, better migration decisions can be made by the algorithm such that the energy consumption is reduced.

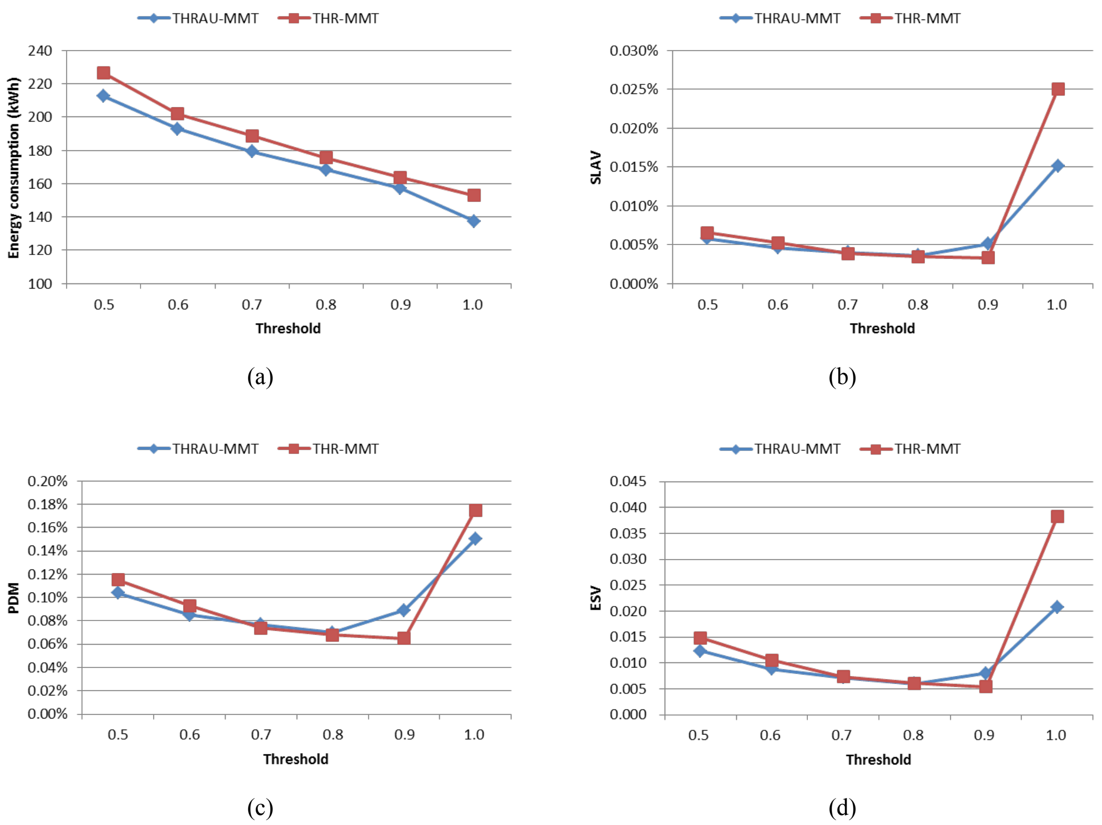

Figure 1: The 4 metrics (Energy consumption, SLAV, PDM and ESV) of THR-MMT and THRAU-MMT algorithms.

5. Conclusion

This paper proposes a novel energy- and QoS-aware virutal machine scheduling algorithm THRAU-MMT. The proposed policy makes use of average utilizations of the VMs on each host to estimate the host's expected utilization. Then, the expected utilization, the actual utilization and the threshold are combined together so as to determine whether a host is overloading more precisely. The simulation results show that compared with other algorithm, the proposed algorithm can effectively reduce energy consumption while guarantee SLAs.

References

[1]M. S.Hasan, E. N.Huh. *Heuristic based Energy-aware Resource Allocation by Dynamic Consolidation of Virtual Machines in Cloud Data Center*[J]. Ksii Transactions on Internet & Information Systems, 2013, 7(8):1825-1842.

[2]M. Dabbagh, B. Hamdaoui, M. Guizani, et al. *An Energy-Efficient VM Prediction and Migration Framework for Overcommitted Clouds* [J]. IEEE Transactions on Cloud Computing, 2016.

[3]A. P. Xiong, C. X. Xu. *Energy Efficient Multiresource Allocation of Virtual Machine Based on PSO in Cloud Data Center* [J]. Mathematical Problems in Engineering, 2014, (2014-6-12), 2014, 2014(5):1-8.

[4]Z. Zhou, Z. Hu, K. Li, et al. *Virtual Machine Placement Algorithm for Both Energy-Awareness and SLA Violation Reduction in Cloud Data Centers* [J]. Scientific Programming, 2016(1):1-11.

[5]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* [J]. Concurrency & Computation Practice & Experience, 2012, 24(13):1397–1420.

[6]A. Beloglazov, R. Buyya. *Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers*[C]. Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science. ACM, 2010:1-6.

[7]Z. Cao, S. Dong. *Dynamic VM consolidation for energy-aware and SLA violation reduction in cloud computing* [C]. International Conference on Parallel and Distributed Computing, Applications and Technologies. IEEE, 2013:363-369.

[8]*The SPECpower benchmark*. http://www.spec.org/power_ssj2008/

[9]A. Beloglazov, J. Abawajy, R. Buyya. *Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing* [J]. Future Generation Computer Systems, 2012, 28(5):755-768.

[10]R. N. Calheiros, R. Ranjan, A. Beloglazov, et al. *CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms* [J]. Software: Practice and experience, 2011, 41(1): 23-50.

[11]K. S. Park, V. S. Pai. *CoMon: a mostly-scalable monitoring system for PlanetLab*. [J]. Acm Sigops Operating Systems Review, 2006, 40(1):65-74.

PoS(ISCC 2017)054