International Journal of Computer & Information Technologies (IJOCIT)
www.ijocit.ir & www.ijocit.org
ISSN = 2345-3877

Dynamic integration of Virtual Machines in the Cloud Computing In Order to Reduce Energy Consumption

Elham Eskandari, Mehdi Afzali*
Islamic Azad University, Zanjan branch[1,2]
Elham.eskandari@yahoo.com[1] *afzali@hacettepe.edu.tr* [2]

Abstract: Cloud computing has been created a revolution in the IT industry by providing enough resources while using it. Profitability of cloud computing had caused to create very large data centers around the world which contain thousand of computing nodes; But these centers consume too much electricity that leads to increase their operating costs and produce a large amount of carbon dioxide too. This research provides new techniques and algorithms for dynamic integration of virtual machines in the cloud computing centers. Our goal is to improve resources' efficiency and reduce energy consumption of cloud computing centers without violating the limitations of desired service quality. Dynamic integration of virtual machines use changes of virtual machines' workload and allocate them nearly to the optimization of physical nodes in order to minimize the number of the woken up physical nodes and consequently save energy. The proposed method is distributed, scalable and efficient in the compromise management between system efficiency and its consumed energy. The most important finding of this study is to provide a distributed method for the dynamic integration of virtual machines and heuristic algorithms.

Keywords: Cloud computing, Virtual machines, Information and communications technology, Energy

1. Introduction

Cloud Computing has created a revolution in the industry of information and communications technology (ICT) by enabling us to reserve computing resources when required based on the concept of payment as usage. An organization can assign its computing needs to a cloud in order to avoid the costs of creating, updating and supporting a private computing infrastructure, or it creates a private cloud for improving the management of its resources and the process of using these resources in different parts of the organization.

The dramatic increase of cloud computing is the result of the establishment of large data centers around the world which each of them consists of several thousand computing nodes. The data centers, however, consume too much electricity resulting in high costs of operation and release of large amount of carbon dioxide to the environment. Consumed energy of data centers had increased to 56% from 2005 to 2010 and it had been between 1.1% and 1.5% of total global electricity consumption in 2010[1]. In addition, the spread of carbon dioxide of information and communications technology industry (ICT) is estimated about two percent of total spreading carbon dioxide (CO2) in the world which is equal to the spread of carbon dioxide in aviation industry [2] and the result of this spread will be greenhouse pollutions. As shown by Koomey [3], the increase of energy consumption will be continued in data centers unless the management of efficient resources has been developed and applied in terms of energy consumption.

It is necessary to eliminate inefficiencies and the number of wasting on the way and the way of giving electricity to computing resources and also the way that these computing resources are used to service to the workload, for solving the issue of excessive use of energy. This task is possible by improving the physical infrastructure of data centers and the management algorithms and allocating of resources. Recent developments in data centers' design have increased the efficiency of physical infrastructures drastically. It has been reported in the open computing project that the Facebook's Oregon data center is reached to 1.08 in terms of effective use [4], and it means that approximately 91% of consumed energy in this data center is belong to computing resources.

The energy wasting resource is hidden in the inefficient use of computing resources. The collected data from over than five thousand servers during six months has been shown that although the majority of servers are usually working, their efficiency hardly reach to 100%. Most of the time, servers work between 10% to 50% of their total capacity which leads to additional cost in reserving more than required resources and consequently more cost in using of these resources [5]. In addition, low efficiency of servers is intensified by the narrow range of their dynamic power: the completely idle servers still consume as much as 70% of their maximum consumption [6]. Hence, the use of servers with low efficiency is very inefficient in terms of energy consumption. This research focuses on efficient management of resources from energy consumption point of view, in cloud data centers. For example, we want to make sure that computing resources are used efficiently in order to give service to the workload to minimize energy consumption while the desired quality of service is met.

2. Dynamic Integration of Virtual Machines Efficiently In Terms of Energy

An ideal way for solving the problem of inefficiency in terms of the energy of implementation of computing system is proportional with energy, it means that the computing power which we obtain, is proportional with the amount of energy that we give to the system [5].This method is partly performed through the technique of dynamic voltage and frequency scaling (DVFS) which is widely implemented in processors. DVFS technique allows us to adjust voltage and frequency of processor dynamically and based on our needs to computing resources. Consequently, the current processors can consume less than 30% of its maximum energy in low function mode, which results to 70% dynamic power consumption range [5].

In contrast, the range of energy consumption of the rest of servers' components is much narrower: 50% for DRAM, 25% for hard disk, 15% for network switches and it is negligible for the rest [6]. The reason is that only the processor supports low-power modes, and other devices can only be turned off partially or completely. However, the overload of transition efficiency from inactive mode to active mode is drastic. For example, the energy consumption of hard disk in completely sleeping mode is negligible but transition to active mode needs a time equal to a thousand times for normal access time. The inefficiency of server components in terms of power consumption in idle mode leads to narrow the range of dynamic power consumption of server to about 30%. This means that even if the server is completely idle, it still has power consumption as much as 70% of its maximum power.

One way to improve using of resources and reducing energy consumption which has been shown to be efficient, is dynamic integration of virtual machines [7, 8, 9, 10, and 11] that are provided by virtualization technology. The virtualization allows clouding providers to run several virtual machines on a single real server, so the efficiency of resources and also the efficiency of investment will be increased. The reduction of energy consumption can be achieved when we switch the idle nodes to low power modes (i.e. sleep or hibernate modes); thus the power consumption is removed in idle mode.

The purpose of this research is dynamic integration of virtual machines and their scheduling for running on the physical machines as energy aware in IaaS clouds and with respect to the limitations of sufficiency. Focusing on IaaS is imposed on us, like several limitations that arise from the nature of these clouds. Solving the issue with mentioned limitations is the main characteristic of our method which distinguishes it from other related works.

Another distinctive aspect of this research from a lot of researches is distributed structure of VMs' management system. Distributed structure of VM management system is essential in large-scale for cloud computing providers, because it enables them to increase the number of servers to thousands nodes. For example, Rackspace cloud as a well-known IaaS provider, manages tens of thousands servers now. In addition, the number of servers is increasing daily as Rackspace has been increased the number of its servers from 82,438 at the end of the first season of 2012 to 84,978 at the end of the second season of 2012 [12]. Another benefit of the distributed structure of management system is that increases the resistance capability of system fault with removing single point of failure, as even if one node or controller goes down, the whole system does not fail.

This study provides an energy aware solution for scheduling and dynamic (integration) of VMs under restrictions of service quality. The results have been simulated based on the simulations of real workloads obtained from the PlantLab VMs.

3. Heuristic Algorithms for the Dynamic Integration of Virtual Machines

In this section, several heuristic algorithms are provided for the dynamic integration of virtual machines based on data analysis of the history of using the VMs from resources. The issue is broken down into four issues below: (1) Detection of the mode that host has been faced with very low workload and requires that all of its VMs to be transferred to other hosts and the host itself must be switched to sleep mode; (2) Detection of the mode that the node has been faced with very high workload and requires that ,one or several of the VMs to be transferred to other nodes in order to reduce the workload of this node; (3) Selecting the VMs that must be transferred from the node with very high workload and (4) finding new nodes for the deployment of selected VMs for migrating in the previous step. The mentioned issues will be examined in the following subsections.

3.1. Detection of the Node with Very Low Workload

Although sophisticated methods can be used for this detection, we will use a very simple way for this task which will have a very good result too. For this task, first the nodes with very high workload are detected by means of related algorithm and their additional VMs are deployed on the destination nodes. Then, the system selects the node which has the lowest efficiency of processor in comparison with the rest of nodes and tries to find the nodes for deploying all of its VMs as the destination node also doesn't face with very high workload. If it could do it, the migration act of VMs on the destination nodes will be done and the source node will be switched to sleep mode. If all of source VMs can't be deployed on other nodes, then the source node will be kept active. This process will be repeated for all nodes that didn't face with very high workload until a node can be reached to sleep mode.

3.2. Detection of Host with Very High Workload

Each computing host runs detection algorithm of very high workload periodically that if it was necessary, some of VMs transfer to other hosts and finally prevent from efficiency reduction and failure of the SLA.

3.3. VM Selection

When a host was detected under circumstance of very high workload, the next step is to select the VMs that must be migrated from this host in order to avoid efficiency reduction. This section provides three policies for VM selection.

3.4 Deployment of Virtual Machines

Deployment issue of virtual machines can be seen as rucksack problem that the rucksacks represent physical nodes and items represent virtual machines. In this problem, the size and price of rucksacks are different due to variability of servers' processing capacity. The size of rucksacks is equal to servers' processing capacity and their price is considered equal to the power consumption of servers. Since the rucksack problem is a NP-

International Journal of Computer & Information Technologies (IJOCIT)
Corresponding Author: *Mehdi Afzali*
April, 2015

Hard problem, it seems logical to use a heuristic algorithm like the best fit decreasing (BFD) for finding the answer. It is proved that this heuristic algorithm does not use more than 11/9 × OPT + 1 rucksacks [15] that OPT is the number of rucksacks that can be obtained by using the optimized algorithm.

Algorithm 1: power aware best fit decreasing algorithm (PABFD)

```
Input: hostList, vmList
Output: vmPlacement
sort vmList in the order of decreasing utilization
for vm in vmList do
    minPower   MAX
    allocatedHost   NULL
    for host in hostList do
        if host has enough resources for vm then
            power = estimatePower(host, vm)
            if power < minPower then
                allocatedHost = host
                minPower = power
    if allocatedHost <> NULL then
        add (allocatedHost, vm) to vmPlacement
return vmPlacement
```

In this section, a change had done in the BFD algorithm and we call the new algorithm as power aware best fit decreasing (PABFD) that is shown in the algorithm 1. First, the algorithm sorts all of VMs decently according to their processor's efficiency and allocates each VM to a host which has the lowest power consumption increase due to adding this VM. This method allows us to gain profit from the heterogeneity of servers in a way that first use servers that has less power consumption. The computing complexity of the algorithm is *nm* in that *n* shows the number of hosts and *m*, the number of VMs which must be deployed.

4. Experiment Setup

Since the studied system is the IaaS system and the IaaS system is a cloud computing environment that is going to create a vision for users that the computing resources are infinite, it seems necessary to evaluate efficiency of the proposed resources' allocation algorithm on the infrastructure of very large data centers. But creation of experiment with large scale and repeatable that is necessary for evaluating and comparing the proposed algorithm is very difficult on the real infrastructure. Therefore, to ensure the repeatability of experiments and also since the real infrastructure is not available, inevitably the simulation has been chosen as the best way to evaluate efficiency.

CloudSim tool [14] had chosen for simulation because this simulator is a new tool which is produced especially for cloud computing environments. Compared to other simulation tools like SimGrid or GangSim, the CloudSim allows making model of virtualized environments supports resources' request when required and also allows for their management.

Simulated data center has 800 heterogeneous physical nodes which half of these servers are from the HP ProLiant ML110 G5 servers and the other half are from the HP ProLiant ML110 G4. The Frequency of servers' processors based on the MIPS is as following: Each server core of HP ProLiant ML110 G4 has a frequency equal to 1860 MIPS and each server core of HP ProLiant ML110 G5 has a frequency equal to 2660 MIPS. The bandwidth of each server is 1Gbps.

The features of the variety of VMs is corresponding with Amazon EC2 VMs, with a difference that all the VMs are single-core and this issue originates from the fact that the used workload for simulation is taken from the single core VMs. The main memory capacity is divided according to the number of cores of each kind of virtual machines: Average samples requiring high processing (2500 MIPS, 0.85 GB), large samples (2000 MIPS, 3.75 GB), small samples (1000 MIPS, 1.7 GB) and tiny samples (500 MIPS, 613 MB). At the beginning, the resources are allocated to them based on the type of VM. During the running, the VMs that use fewer resources will be creating an opportunity to be able to integrate them dynamically.

4.1 Efficiency Criteria

Several efficiency criteria have been used in order to evaluate and compare the efficiency of algorithms. One of the criteria is the total energy consumed by the servers of data center. The criteria which are used for examination of fault amount of the SLA are SLAV, OTF and PDM. Another criterion of efficiency is the number of performed migration due to the integration of virtual machines.

The main criteria of efficiency are total amount of energy consumption and SLAV that correlated with each other negatively so that energy consumption is reduced by increasing of SLAV. The aim of resources' management is to minimize both the energy consumption and SLAV criteria.

4.2. Workload

It is necessary that experiments are done based on the obtained workload of a real system in order to be able to rely on the archived results of simulation. Provided data of CoMon project is used for the experiments of this chapter which is the supervision infrastructure for PlantLab [13].

Table 1: features of the workload data (processor efficiency)

Date	Number of VMs	Average (%)	Standard deviation (%)	First quarter (%)	Mean (%)	Third quarter (%)
3/3/2013	1052	12.31	17.09	2	6	15
6/3/2013	898	11.44	16.83	2	5	13
9/3/2013	1061	10.70	15.57	2	4	13
22/3/2013	1516	9.26	12.78	2	5	12
25/5/2013	1078	10.56	14.14	2	6	14
3/4/2013	1463	12.39	16.55	2	6	17
9/4/2013	1358	11.12	15.09	2	6	15
11/4/2013	1233	11.56	15.07	2	6	16
12/4/2013	1054	11.54	15.15	2	6	16
20/4/2013	1033	10.43	10.43	2	4	12

Data include processor efficiency by more than thousand VMs from the servers that are located in more than 500 places around the world and are selected randomly in ten days of March and April 2013. The time interval of measuring efficiency is five minutes.

The daily data features are shown in Table 1. The data of the sentence which came in the second chapter prove: efficiency mean of processor is below 50%. During the simulation, the workload related to one VM from daily related VMs is allocated to each VM. Since the purpose of the experiments is to test integration algorithm, memory limitation is not considered in the integration of virtual machines.

4.3. Simulation Results and Analyzing Them

All of combinations of five proposed algorithms for the detection of very high workload situations (THR, IQR, MAD, LR and LRR) with three VM selection algorithms for migration (MMT, RS and MC) have been simulated by means of the workload data described in Section 3.4.3. In addition, the parameters of each detection algorithm of very high workload has changed as below: for THR from 0.6 to 1.0 with an increase of 0.1 in each step, for IQR and MAD from 0.5 to 3.0, with an increase of 0.5 in each step, for LR or LRR from 1.0 to 1.4 with an increase of 0.1 in each step. These changes result in 81 combinations of algorithms and parameters.

According to normality test of Ryan-Joiner, the amounts produced from the combination of algorithms for ESV criterion does not follow a normal distribution with P-value <0.01. Thus, the mean value of ESV criterion is used to compare different combinations of algorithms and select parameter for a specific combination that minimizes the computed ESV mean during ten days workloads. The achieved results are shown in figure 1.

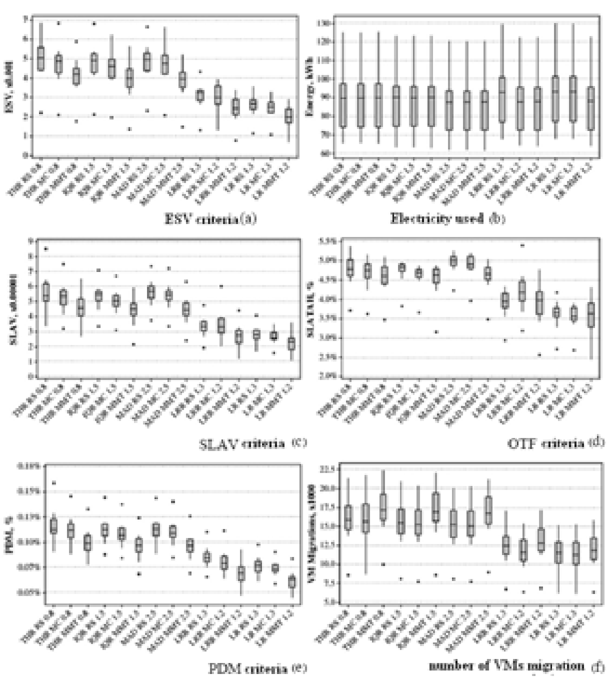

Figure. 1: comparison of different algorithm in terms of PDM, OTF, SLAV, ESV criteria, energy consumption and the number of performed migrations

According to the normality test of Ryan-Joiner, the values generated for ESV criterion follow a normal distribution with P-value> 0.1. Three pairs T-test for detecting selection policy of VM had done that minimize the ESV criterion among all combinations and the results are shown in Table 2.

Table 2: The comparison of VM selection policies by means of T-test pair

Policy 1 (ESV × 10^{-3})	Policy 2 (ESV × 10^{-3})	Difference (× 10^{-3})	P-value
RS (4.03)	MC (3.83)	0.196 (0.134, 0.258)	P-value < 0.001
RS (4.03)	MMT (3.23)	0.799 (0.733, 0.865)	P-value < 0.001
MC (3.83)	MMT (3.23)	0.603 (0.533, 0.673)	P-value < 0.001

The T-tests showed that the use of MMT policy statistically decreases the ESV criterion value with P-value <0.001 drastically. Then combinations of detection algorithms of very high workload were analyzed by MMT policy.

To meet the hypotheses of ANOVA model, ESV criterion values were converted by means of the square root function for different combinations of detection algorithms of very high workload with MMT policy. The Ryan-Joiner test performed successfully with P-value> 0.1 on standardized remainders generated from data conversion that confirms this hypothesis that samples had been taken from a normal distribution. The diagram obtained from the standardized remainders against processed values showed that the equal variance hypothesis is correct.

With respect to the fact that hypotheses of the model were correct, we used F-Test for finding that whether there is significant difference between the results produced from combinations of detection algorithms of very high workload with MMT policy and selected parameters or not. The test results showed that there is significant difference between the results with P-value <0.001. Tukey's pairwise comparisons are summarized in Table 3.

Table 3, Turkey's pairwise comparisons by means of ESV, there is significant difference between the related values of different groups.

Group	Confidence level 95%	Square root of ESV with accuracy of two decimal places	Policy
A	(5.70, 6.98)	6.34	THR-MMT-0.8
A	(5.44, 6.87)	6.16	IQR-MMT-1.5
A	(5.49, 6.77)	6.13	MAD-MMT-2.5
B	(4.22, 5.41)	4.82	LRR-MMT-1.2
B	(3.83, 4.91)	4.37	LR-MMT-1.2

There isn't significant difference between obtained results from THR-MMT-0.8, IQR-MMT-1.5 and MAD-MMT-2.5 algorithms (group A) according to Turkey's pairwise comparisons. Also the results showed that there isn't significant difference between the obtained results from LRR-MMT-1.2 and LR-MMT-1.2 algorithms (group B). But there is significant difference between the algorithms based on local regression and other algorithms. However, a performed T- test pair to compare generated means for ESV criterion by LRR-MMT-1.2 and LR-MMT-1.2 algorithms showed that there is significant difference between them with P-value < 0.00 statistically. The mean difference with 95% confidence level (3.23×10^{-4}, 5.19×10^{-4}) is equal to 4.21×10^{-4}.

Since the T-test pair produces more accurate results than Turkey's pairwise comparisons, it can be concluded that the LR-MMT-1.2 algorithm produces the best result in term of ESV criterion among all algorithms. In addition, the trade-off between energy consumption and violation rate from SLA can be adjusted by changing safety margin parameter of LR algorithm. The combinations' results of each very high workload detection algorithm with the best parameter and MMT policy along with benchmark algorithms are shown in table 3.5. Benchmark policies include Non Power Aware (NPA), DVFS and THR combined with MMT policy. The NPA policy forces all of the hosts to work with highest power in all times.

Several results can be taken from the simulations according to obtained observations as below:

Table 5: The simulations' results of the best algorithm combinations and benchmark algorithms (mean values)

policy	ESV ($\times 10^{-3}$)	Energy (kwh)	SLAV ($\times 10^{-5}$)	OTF (%)	PDM (%)	Numbr of migration $\times 10^{-3}$
NPA	0	241.9	0	0	0	0
DVFS	0	613.6	0	0	0	0
THR-MMT-1.0	20.1	75.36	25.78	24.97	0.10	13.64
THR-MMT-0.8	4.19	89.92	4.57	4.61	0.10	17.18
IQR-MMT-1.5	4.00	90.13	4.51	4.64	0.10	16.93
MAD-MMT-2.5	3.94	87.67	4.48	4.65	0.10	16.72
LRR-MMT-1.2	2.43	87.93	2.77	3.98	0.07	12.82
LR-MMT-1.2	1.98	88.17	2.33	3.63	0.06	11.85

- Algorithms of dynamic integration of virtual machines work better than static algorithms such as NPA and DVFS with significant difference.
- Very high workload detection algorithms based on statistic drastically work better than static algorithms based on threshold of processor efficiency in term of reducing violation rate from SLA.
- The MMT policy produces better results than MC and RS policies and it means that the minimizing of migration time is better than minimizing of correlation between VMs of a physical node.
- Very high workload detection algorithms based on local regression work better than algorithms based on threshold in term of reducing violation rate from SLA and reduction of the number of migrations.
- The LR algorithm produces better results than its steady match that it can be justified due to the fact that showing the fast reaction to increases of workload is more important than unifying the unrelated observations for simulated workload.

In simulations, the average time that takes until the node is switched to sleep mode is equal to 1933 seconds for LR-MMT-1.2 algorithm and it is with 95% confidence level in the interval (1740 to 2127). This means that a node averagely after 32 minutes activity will be switched to sleep mode. Since the new servers support the transitions with very low delay which consume less power in sleep mode, this value is effective for real systems. It is shown in [16] that a blade server which works with total capacity uses 45 V power but this number is reduced to 10 V in sleep mode. In this kind of server, the transition delay between different modes is 300 milliseconds.

The average number of servers which switch in sleep mode is up to 1272 servers daily (The total number of servers is 800.) and it is with 95% confidence level in the interval (1211 to 1333). For this combination of algorithms, the average time which takes to migrate a virtual machine from a host is 15.3 seconds and it is with 95%confidence level in the interval (15.2 to 15.4). The average running time of LR-MMT-1.2 algorithm on an Intel Xeon 3060 PC with 2.4 GHz processor and 2 G memories was 0.2 seconds and it was with 95%confidence level in the interval (0.25 to 0.15).

5. Conclusion

The providers of cloud computing need to use strategies of energy aware resources' management like integration of virtual machine dynamically and switch off the idle servers in order to save energy for maximizing of their return on investment. But these integrations are not very simple because it may lead to violation from SLA that if it happens, the cloud computing provider should pay a fine. In this chapter, the distributed heuristic algorithms were provided for integration of virtual machines based on analyzing historical data using resources by them and capability of predictability of workload.

The most important findings of this study are:

1- Providing a distributed method for dynamic integration of virtual machines which is consisted of four sections.
   a) Detection of the node with very low workload for migrating of all running virtual machines on the node and switching off the related node in order to save energy consumption
   b) Detection of the node with very high workload for migrating of some running virtual machines on the node in order to prevent from violating quality limitations of desired service.
   c) Detection of virtual machines which shall be migrated over the node with very high workload.
   d) Detection of the nodes which can accept virtual machines that migrate over the nodes with very high or low workloads.

2- Providing heuristic algorithms for each of above four mentioned sections.

The simulation results of large data centers by means of obtained workloads from more than thousands PlanetLab virtual machines showed that the proposed detection algorithm of very high workload based on local regression along with policy of selecting MMT virtual machine had shown better performance significantly in terms of ESV criterion due to drastic reduction of violation rate from SLA and the number of migrations.

The most important disadvantage of proposed algorithm is its inability to determine the limitation of quality of service (QoS) explicitly. The efficiency of algorithm can only be adjusted indirectly through the change of algorithm parameter with respect to QoS.

References

[1] *Growth in data center electricity use 2005 to 2010.* s.l.: Analytics Press.

[2] Gartner estimates ICT industry accounts for 2 percent of global CO2 emissions. [Online] 2007. [Cited: 1 17, 2013.] http://www.gartner.com/it/page.jsp?id=503867.

[3] Koomey, J. G. *Estimating total power consumption by servers in the US and the.* s.l. : Lawrence Berkeley National Laboratory, 2007.

[4] Project, Open Compute. Energy efficiency. [Online] [Cited: 11 21, 2012.] http://opencompute.org/about/energy-efficiency/.

[5] *The case for energy-proportional computing*. Holzle, L. A. Barroso and U. 12, s.l. : Computer, 2007, Vol. 40, pp. 33-37.

[6] *Power provisioning for awarehouse-sized computer*. X. Fan, W. D. Weber, and L. A. Barroso. s.l.: the 34th Annual International Symposium on Computer Architecture (ISCA), 2007.

[7] *Entropy: A consoli-dation manager for clusters*. F. Hermenier, X. Lorca, J. Menaud, G. Muller, and J. Lawall. s.l. : the ACM SIGPLAN/SIGOPS Interna-tional Conference on Virtual Execution Environments (VEE), 2009.

[8] *Power and performance management of virtualized computing environments via lookahead control*. D. Kusic, J. O. Kephart, J. E. Hanson, N. Kandasamy, and G. Jiang. 1, s.l. : Cluster Computing, 2009, Vol. 12.

[9] *VirtualPower: Coordinated power management in vir-tualized enterprise systems*. Schwan, R. Nathuji and K. 6, s.l.: ACM SIGOPSOperating Systems Review, 2007, Vol. 41.

[10] *No "power" struggles: Coordinated multi-level power management for the data center*. R. Raghavendra, P. Ranganathan, V. Talwar, Z. Wang, and X. Zhu. 1, s.l. : SIGARCH Computer Architecture News, 2008, Vol. 36.

[11] *pMapper: power and migration cost aware application placement in virtualized systems*. A. Verma, P. Ahuja, and A. Neogi. s.l.: the 9th ACM/IFIP/USENIX International Conference on Middleware, 2008.

[12] Rackspace, US Inc. Rackspace hosting reports second quarter 2012 results. [Online] 2012. [Cited: 11 06, 2013.] Available: http://ir.rackspace.com/phoenix.zhtml?c=221673&p=irol-newsArticle&ID=1723357.

[13] *CoMon: a mostly-scalable monitoring system for Planet-Lab*. Pai, K. S. Park and V. S. 1, s.l. : ACM SIGOPS Operating Systems Review, 2006, Vol. 40.

[14] *CloudSim: A toolkit for modeling and simulation of Cloud computing environ-ments and evaluation of resource provisioning algorithms*. R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. D. Rose, and R. Buyya. 1, s.l. : Software: Practice and Experience, 2011, Vol. 41.

[15] *A simple proof of the inequality FFD (L)< 11/9 OPT (L)+ 1,for all l for the FFD bin-packing algorithm*. Yue, M. 4, s.l.: Acta Mathematicae Applicatae Sinica (English Series), 1991, Vol. 7.

[16] *PowerNap: eliminating server idle power*. D. Meisner, B. Gold, and T. Wenisch. 3, s.l. : ACM SIGPLAN Notices, 2009, Vol. 44.