Resource Allocation and Server Consolidation Algorithms for Green Computing

Mostafa Sami, M. Haggag, Dina Salem

Abstract— In cloud computing data centers, the only interest was high performance without paying much attention to energy consumption that is growing rapidly. Many huge problems come up from this high energy consumption. Turning green is a new concept for data centers, to solve these problems. Green computing means using resources efficiently and eco-friendly. This research paper proposes a scalable system that helps data centers use energy in an efficient way, by combining a resource allocation algorithm and a server consolidation one, their goal is to minimize the number of physical machines used to execute all required tasks.

Index Terms— CPU utilization; energy consumption; green computing; physical machine; virtual machine migration; server consolidation; threshold.

1 INTRODUCTION

Clouds are a large pool of available virtualized and physical resources (such as CPU, storage, network bandwidth, and so on). These resources can be dynamically reallocated to adjust to a changeable load, permitting also for most advantageous resource utilization [1].

Until recently, high performance has been the only interest for data centers, and this demand has been satisfied without paying much attention to energy consumption, that is growing rapidly. While there are huge problems that come up from high energy consumption, starting from high energy bills in data centers, to raise of environmental concerns and increase of system failures. That's why infrastructure providers are under massive pressure to decrease the consumption of energy, the goal is not only to reduce data centers' energy cost, but also to meet governmental rules and environmental standards [2, 3].

In regards to these facts, showing the importance of the reduction of energy consumption; a new concept called "Green Computing" is recently invented. Green computing points to environmentally sustainable computing. It's the study and implementation of using computing resources in an efficient and eco-friendly way [4]. Its objectives include 1) improving energy efficiency and power management practices, 2) increasing hardware utilization efficiency, 3) reducing life cycle costs, and 4) looking for ways to cut down on computer wastes [5].

These objectives can be approached from many directions.

1) Energy Efficient Hardware Architecture, which enables decreasing CPU speeds and turning off some of the hardware components. 2) Energy-Aware Job Scheduling.

In addition to two other ways to reduce power consumption

3) Server Consolidation,(i.e. turning off unused machines), and 4) Energy-Efficient Network Protocols and Infrastructures, which is a recent research.

An important key in all the above methods is to realize a good compromise between energy efficient consumption and application performance.

This research paper is concerned to help data centers turning green by implementing an optimum resource allocation algorithm together with a server consolidation algorithm where their objective is to use the minimum number of physical machines that can host the requests sent to data centers, considering performance degradation and SLA violation. The allocation algorithm runs from the system initial state – where no hosts are allocated yet to any request, then the server consolidation algorithm is executed whenever a host having a request completed.

Section 2 contains a survey study for the related work in the field of energy efficiency. In Section 3, there is a description of the proposed system, its input and the pseudo code of the applied algorithms. Experiments' environments, results and comparisons are found in Section 4. In Section 5, readers will find the research conclusion.

2 RELATED WORK

Saini and Indu [6] defined a middle layer between the cloud servers and the client's requests that will perform the allocation of the processes to multiple clouds in overload and underload conditions. As the request is performed by the user, certain parameters are defined with each user request (like the arrival time, process time, deadline, input output specifications). Resource allocation is performed sequentially, and each process must be executed within the deadline. If more than one process must be executed at the same time, so process migration from a cloud to another takes place.

Suchithra and Rajkumar [7] proposed an algorithm for server consolidation that minimizes the number of physical servers, by packing the jobs in the existing servers that bears the heaviest workload (CPU utilization). The aim is not only to keep minimum physical servers but also to reduce the number of migrations. They concentrated only on the migration concept, leaving other issues, like jobs' completion time, migration time or costs, which may be considered in their future work.

Marzolla, Babaoglu and Panzieri [8] proposed a gossip-based algorithm called V-Man, for consolidation of virtual machines that means maximizing the number of empty hosts. This algorithm is fully decentralized and doesn't require any global knowledge. Each server exchanged messages with a limited number of peers; these messages are used to i) maintain an unstructured overlay network, and ii) exchange VMs from lightly loaded nodes to heavily loaded ones. After each round, V-Man produces a new allocation which quickly converges toward the optimal one. It's implemented using a simulator called Peersim. They assume that all VMs are identical and didn't take into consideration a cost model to choose a certain VM to be migrated.

Murtazaev and Oh [9] proposed an algorithm for server consolidation called Sercon. This algorithm has two objectives: minimizing number of nodes as well as number of migrations. In this algorithm all VMs and nodes have a calculated score representing their loads. The CPU utilization and the memory are considered for representing the load. A threshold value is chosen that determine if a node is most loaded or least loaded. From the least loaded node, the list of VMs is ordered decreasingly, then all these VMs are tried to be migrated to the most loaded nodes. Migration of VMs is done if it results in the release of a node.

Beloglazov and Rajkumar [10, 11] proposed a novel technique for the energy- efficient threshold-based dynamic consolidation of virtual machines VMs with auto-adjustment of the threshold values. It's an approach for dynamic adaption of allocation of VMs in run-time applying live migration according to current utilization of resources. It can effectively handle strict Quality of Service (QoS) requirements, heterogeneous infrastructures and VMs. The software system architecture is tiered comprising a dispatcher, global and local managers.

3 PROPOSED SYSTEM

There are four kinds of resources that are provided in data centers, 1) CPU, 2) disk, 3) memory and 4) network bandwidth [7]. It's considered here, in this research work, only CPU, while energy consumption scales linearly with CPU utilization [12].

Virtualization is a keystone for cloud computing so as to green computing [9]. It replaces the old concept of "one server one application model", where multiple virtual machines can run in one physical server. Even one virtual machine can run more than one user request [7].

The proposed system, combining two techniques together, resource allocation and server consolidation has a limited number of servers or physical machines PMs that are switched to the sleep mode to save energy until a virtual machine VM is assigned to them.

The goal is to use the least number of PMs that are enough to run a v number of VMs in a dynamic system, while no prior knowledge of VMs usage during the system run, considering SLA violation and performance degradation. It's supposed when running the two algorithms that each VM is combined to only one request, that is using space shared requests allocation.

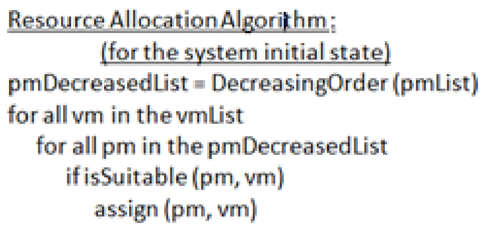
The resource allocation algorithm is executed first, for mapping all requests sent to the data center over the least number of PMs. Then the server consolidation algorithm is executed in the case of any VM release, to guarantee continuous energy efficient use of the system resources. Also minimizing the number of migrations is another vital goal taken into the system's consideration, while migration process takes time and has cost that affects system performance, represented in task completion times.

The SLA is preserved by applying a static threshold, so no PM is found over-utilized [13]. The threshold value used is 85% from the CPU utilization [14].

3.1 Resource Allocation

The resource allocation problem is related to general packing problems. The bin-packing problem is one of those problems, VMs are considered as objects and PMs as bins. The bin packing problem is known as an NP-hard, and a number of heuristic algorithms are offered to give sub-optimal solutions to such problems, like Next Fit (O(n)), First Fit (O(n log n)) and Best Fit (O(n log n)) [9].

There are a p number of PMs that are to be allocated to a v number of VMs. Firstly the PMs are decreasingly ordered depending on their capacity, measured in Million Instructions per Second, MIPS, as to let the powerful machine allocate the maximum number of VMs, but don't exceed the 85% of the machine MIPS. Otherwise the allocation goes through the next machine in the decreased list having free spaces for the VM in use. Figure 1 shows the algorithm of the resource allocation.



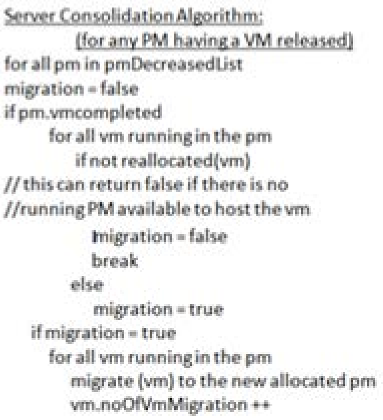
```
Resource Allocation Algorithm:
        (for the system initial state)
pmDecreasedList = DecreasingOrder (pmList)
for all vm in the vmList
    for all pm in the pmDecreasedList
        if isSuitable (pm, vm)
            assign (pm, vm)
```

Fig 1. The resource allocation algorithm pseudo-code

3.2 Server Consolidation

Server consolidation is an effective technique to increase the utilization of resources while decreasing the energy consumption in a Cloud computing environment. To apply this technique, live VM migration technology is used; it combines VMs existing on multiple under-utilized servers onto a single server, so that the rest of servers can be put to an energy-saving state [15].

In the proposed system, this algorithm is executed periodically to check for any PM having a VM completed and released, once found, all other VMs residing in this PM are checked for migration, if all VMs can be re-allocated to other running PMs, so migrations are done in order to switch this machine off, as to minimize the energy consumption. The number of migrations is counted for each VM, and it's taken into consideration. Figure 2 shows the server consolidation algorithm pseudo-code.



```
Server Consolidation Algorithm:
        (for any PM having a VM released)
for all pm in pmDecreasedList
migration = false
if pm.vmcompleted
      for all vm running in the pm
            if not reallocated(vm)
//this can return false if there is no
//running PM available to host the vm
                  migration = false
                  break
            else
                  migration = true
   if migration = true
      for all vm running in the pm
            migrate (vm) to the new allocated pm
            vm.noOfVmMigration ++
```

Fig 2. The server consolidation algorithm pseudo-code

4 Experimental Results

To run the system, CloudSim toolkit has been chosen as a simulation framework for cloud computing environment. It is commonly used in many research papers; it supports modeling of on-demand virtualization enabled resource and the management of applications [16].

Our system has been run and compared with two other algorithms implemented in the Cloudsim package. The first class is a simple VM allocation policy that does not apply any optimization of the VM allocation (SMP) and a Static Threshold (THR) VM allocation policy [17].

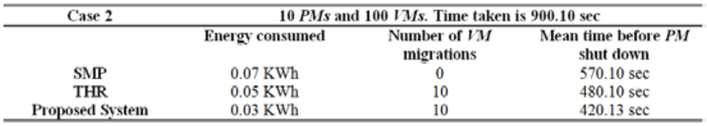
The experiment is executed in four different cases. In the first case, the simulated data center consists of 5 PMs and 20 VMs. In this case, the time taken to execute all jobs is 900.1 sec; the energy consumed by the three algorithms – SMP, THR and the proposed system – is the same, 0.2 KWh. This is resulted because the number of PMs is not large that they run almost all the jobs without having extra switched off PMs. But the number of migrations in the THR algorithm is double the same number in the proposed sytem.

In the second case, it consists of 10 PMs and 100 VMs, shown in Table 1. The third case consists of 50 PMs and 500 VMs, shown in Table 2. The last case consists of 100 PMs and 1000 VMs, shown in Table 3. Each PM is modeled to have only one CPU. PMs' performance is equivalent to 1860 or 2660 MIPS, 4 Mb of RAM, and 1 GB of storage. Each VM requires one CPU core with 50, 100, 200 or 250 MIPS. RAM and storage requests for VMs are minimized while they are out of research. Also all requests are of length equal to 450 Million Instruction MI. The threshold used for the THR and the proposed system is 85%.

The simulated results for the three cases are presented in the following tables.

Table1

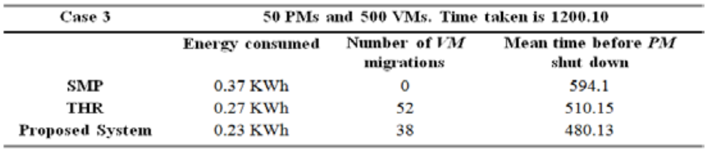
Comparison Results for the Second Case



Case 2	10 *PMs* and 100 *VMs*. Time taken is 900.10 sec		
	Energy consumed	Number of *VM* migrations	Mean time before *PM* shut down
SMP	0.07 KWh	0	570.10 sec
THR	0.05 KWh	10	480.10 sec
Proposed System	0.03 KWh	10	420.13 sec

Table2

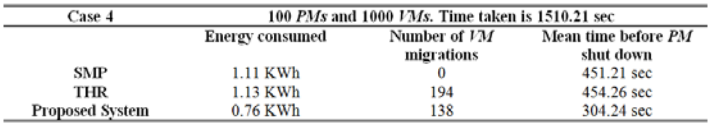
Comparison Results for the Third Case



Case 3	50 PMs and 500 VMs. Time taken is 1200.10		
	Energy consumed	Number of *VM* migrations	Mean time before *PM* shut down
SMP	0.37 KWh	0	594.1
THR	0.27 KWh	52	510.15
Proposed System	0.23 KWh	38	480.13

Table3

Comparison Results for the Fourth Case



Case 4	100 *PMs* and 1000 *VMs*. Time taken is 1510.21 sec		
	Energy consumed	Number of *VM* migrations	Mean time before *PM* shut down
SMP	1.11 KWh	0	451.21 sec
THR	1.13 KWh	194	454.26 sec
Proposed System	0.76 KWh	138	304.24 sec

From the above results, it's found that in the first case, where the data center is small, the three algorithms have almost same energy consumed. But in the other three cases, where the data center is larger, there will be a difference in the energy consumed, whereas the proposed system proves better energy utilization; also this difference increases with the increase of the number of PMs and VMs. About the number of migrations, the SMP is not taken into consideration for this comparison, while there is no optimization and no VM re-allocations are done. The proposed system did fewer migrations than the THR algorithm in more cases, which makes the system more stable and minimizes the migration costs. The third and last comparison column is for the time taken for the first PM shutting down. Also it's shown that the proposed system takes less time to switch off the first PM.

The figures 3, 4 and 5 are the visual representation of the four cases. It's shown that the proposed algorithm proves better performance in larger environments, which makes it scalable and suitable for real clouds.

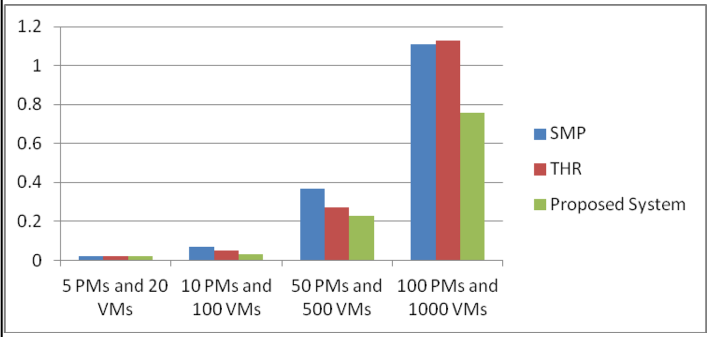


Fig 3. Shows the energy consumed

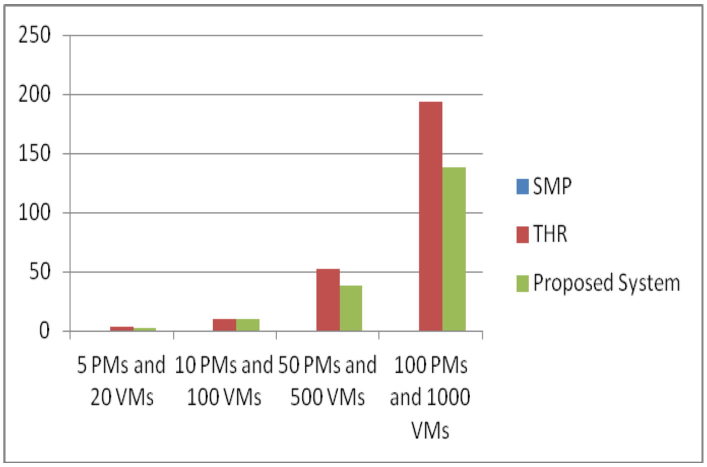


Fig 4. Shows the number of migrations

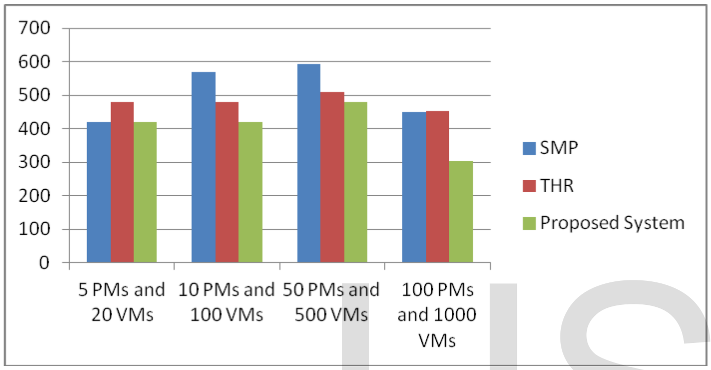


Fig 5. Shows the mean time before PM shut down

5 Conclusion

The proposed system helps data centers turn green by applying a resource allocation algorithm then a server consolidation one. Both algorithms aim to allocate the minimum number of PMs to all VMs in the system. The resource allocation algorithm starts running from the system initial state, and the server consolidation is executed once a PM has a VM released, such PM can be switched off if all of its VMs can be reallocated to other running PMs. This can be shown from the results tables where the mean time before PM shut down is always the smallest. Also the proposed system proves that is suitable for real clouds due to its scalable feature.

References

[1] Jiyi WU, Lingdi PING, Xiaoping GE et al, “*Cloud Storage as the Infrastructure of Cloud Computing*,” International Conference on Intelligent Computing and Cognitive Informatics. China, 2010 IEEE

[2] Anton Beloglazov, Jemal Abawajyb, Rajkumar Buyya, “*Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing*,” Future Generation Computer Systems 28 (2012) 755–768

[3] Xiaomin Zhua, Rong Geb, Jinguang Sunc et al, “*3E: Energy-Efficient Elastic Scheduling for Independent Tasks in Heterogeneous Computing Systems*,” The Journal of Systems and Software 86 (2013) 302–314

[4] S.V.S.S. Lakshmi, Ms. I Sri Lalita Sarwani, M.Nalini Tuveera, “*A Study On Green Computing: The Future Computing And Eco-Friendly Technology*,” International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue4, July-August 2012, pp.1282-1285

[5] Miss Swati Aggarwal, Mrs. Monika Garg, Mr. Pramod Kumar, “*Green Computing is SMART COMPUTING – A Survey*,” International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 2, Issue 2, February 2012, www.ijetae.com

[6] Ranjana Saini and Indu, “*Efficient Job Scheduling of Virtual Machines in Cloud Computing*,” International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 9, 2013

[7] R. Suchithra, N. Rajkumar, “*Efficient Migration – A Leading Solution for Server Consolidation*,” International Journal of Computer Aplications, Volume 60 – No. 18, December 2012

[8] Moreno Marzolla, OzalpBabaoglu and Fabio Panzieri, “*Server Consolidaton in Clouds Through Gossiping*,” World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium

[9] Aziz Murtazaev and Sangyoon Oh, “*Sercon: Server Consolidation Algorithm using Live Migration of Virtual Machines for Green Computing*,” IETE Technical Review, Vol 28, Issue 3, 2011

[10] Anton Beloglazov and Rajkummar Buyya, “*Energy Efficient Resource Management in Virtualized Cloud Data Centers*,” 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010

[11] Anton Beloglazov and Rajkumar Buyya, “*Adaptive Threshold-Based Approach for Energy-Efficient Consolidation of Virtual Machines in Cloud Data Centers*,” Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science (MGC 2010), Bangalore, India: ACM, 2010

[12] Awada Uchechukwu, Keqiu Li, Yanming Shen, “*Energy Consumption in Cloud Computing Data Centers,*” International Journal of Cloud Computing and Services Science (IJ-CLOSER) Vol.3, No.3, June 2014 ISSN: 2089-3337

[13] Anton Beloglazov and Rajkumar Buyya, “*Adaptive Threshold-Based Approach for Energy-Efficient Consolidation of Virtual Machines in Cloud Data Centers*,” Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science (MGC 2010), Bangalore, India: ACM, 2010

[14] X. Zhu, D. Young, B. J. Watson et al., “*1000 Islands: Integrated capacity and workload management for the next generation data center*,” the 5th International Conference on Autonomic Computing (ICAC), 2008, pp. 172– 181

[15] Qi Zhang, Lu Cheng, Raouf Boutaba, “*Cloud Computing: State-of-the-Art and Research Challenges*,” the Brazilian Computer Society 2010

[16] Rajkumar Buyya, Rajiv Ranjan and Rodrigo N Calheiros, “*Modeling and Simulation of Scalable Cloud Computing Environments and the Cloudsim Toolkit: Challenges and Opportunities*,” International Conference on High Performance Computing & Simulation, HPCS '09, 2009

[17] Anton Beloglazov, and Rajkumar Buyya, “*Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers*,” Concurrency and Computation: Practice and Experience (CCPE), Volume 24, Issue 13, Pages: 1397-1420, John Wiley & Sons, Ltd, New York, USA, 2012