International Journal of Latest Trends in Engineering and Technology
pp.017-025
DOI: http://dx.doi.org/10.21172/1.IRES.02
e-ISSN: 2278-621X

ENERGY-EFFICIENT CLOUD COMPUTING WITH TASK CONSOLIDATION

Parthasarathi pattnayak[1] and Pamela pal[2]

Abstract- Energy saving in a Cloud Computing environment is a multidimensional challenge, which can directly decrea the in-use costs and carbon dioxide emission, while raising the system consistency. We in this paper, suggest the ener saving with task consolidation, by minimizing the number of unused resources in a cloud computing environment. In t paper, it is proposed that task consolidation using Minimization Of IDLE VM algorithm. The existing ECTC has be compared with the recital of the recommend algorithms, and MaxMaxUtil energy conscious task consolidati algorithms (i.e. it minimizes the number of idle resources by allocating a task at an instance to a VM which is curren idle). The outcomes have shown that the suggested algorithms surpass the ECTC and MaxMaxUtil algorithm in terms the CPU utilization and energy consumption.

Key words: CPU utilization, Energy consumption, Idle VM, Task consolidation, Virtualization VMs.

I. INTRODUCTION

The development of technologies like network devices, hardware capacities and software applications have made cloud computing a popular computing paradigm. In all these computing systems the resources can be dispersed widely and the taking part of resources can range from numerous physical servers to an intact data centre. Proficient methods of management are required to join them together and make these resource use optimally at various scales[17]. Therefore, in recent years the focus of research has shifted to exploit resources efficiently for minimizing the power consumption by data centers. Virtualization is one of the key technologies in cloud computing to provision Virtual Machines (VMs) dynamically, as per the dynamic requirements on the physical machines. Therefore, numerous techniques have been proposed and developed which have improved the resource utilization in terms of task allocation, memory firmness, defining threshold for use of resources and request inequity among VMs. There have been several research studies which tries to improve the resource utilization as well as to reduce the energy consumption in different conditions. But there is need of a substitution between the entities. The overall goal of both the situations is to reduce the cost for data centers. The power consumption is directly proportional to the energy consumption. Utilization of CPU is related to energy consumption, as the utilization of CPU is directly proportional to the energy consumption. However, if there is a higher CPU utilization of a system, that does not equate to energy efficiency with in that system. Thus, this process has motivated the idea to maintain a threshold level for CPUs with high levels of utilization for energy saving. Infrastructureas-a-Service (IaaS), Platform-as-a-Service (PaaS), Softwareas-a-Service (SaaS) and Database-as-a-Service (DaaS) are the four levels of access in which clouds are deployed for the clients. The task has originated by the different type of customers according to their requirements. There are several heuristic algorithms proposed for local cloud for the centralized controller which has been power aware. Based on the system structure and the characteristics of the cloud infrastructures, a function between the resources of cloud and the combinatorial allocation task has been proposed, as an economics

---

[1] *School of Computer Application KIIT University, Bhubaneswar, Odissa, India*
[2] *Cognizant, Kolkata, West Bengal,India*

based optimization model. VMs have been designed to run on various servers which provide the multiple Operating system environments for different applications. Particularly, executing an application which requires resources has been made available for resource provisioning and VM provisioning. Resource provisioning is scheduling the requests for the physical resources where-as VM provisioning creates the instance of VMs as required by the different applications [3]. Server or workload consolidation is the main aim of the task consolidation problem. It allows the servers on a single physical server for minimization of energy consumed by a cloud data center. In the present paper, the task consolidation problem has been addressed to allocate n tasks to a place of different sources and the utilization of nodes and distributed VMs are maintained by energy efficiency and load management. In this paper, the greedy heuristic algorithm has been evaluated and implemented for three basic task consolidations which assign tasks to the physical servers for minimizing the total energy consumed. The proposed heuristic is to minimize the no.(number thereafter) of idle Virtual Machines and minimize the no. of idle Virtual Machines on the way to as minimum as possible. It has also been shown that the performance improvement is based on different tasks.

II. CLOUD SYSTEM MODEL

The current section depicts the cloud and its function with the energy models. It also defines the job consolidation problem. The high level architecture of the cloud system is shown in figure 1 [16]. Virtualization allows the cloud providers to create a set of VMs on a single physical machine that improves the ROI (Return On Investment). The EC(energy consumption thereafter) may be reduced with switching off the nodes which is idle, and eliminates the idle power consumption of the given system [11-17].

In the present work, the target system has been used r resources for a set N which can be connected within the logic that a common direction lives between whichever individual sources as shown in Figure 2. It assumes that the sources are identical in expressions of their potential of computing. The virtualization technologies justifies this. The present study has however not considered the federated cloud environment in which the data centers can be placed at different physical locations and the client requests may be processed at various geographical locations. Several services which have been offered by cloud providers can be in the form of DaaS, SaaS, PaaS and IaaS. When instances of these desired distinct services are running over the different platforms, they may be termed as computational tasks. Applications consist of several tasks which are allocated to the different resources. IaaS is responsible for requests with predefined point in time frame i.e. pay-per-hour, whereas DaaS, SaaS and PaaS may be frequently not stalwartly joined through an unchanging quantity of instance i.e. pay-per-utilize. On the other hand, it can live to have an estimate to check desires for SaaS, PaaS and DaaS based on past data. It assumes that the CPU utilizations of each and every service request can be classifiable according to the requirements. It has also been presumed to facilitate the memory & disk exploit associates by processor utilization in which service, task and applications are used interchangeably [6].

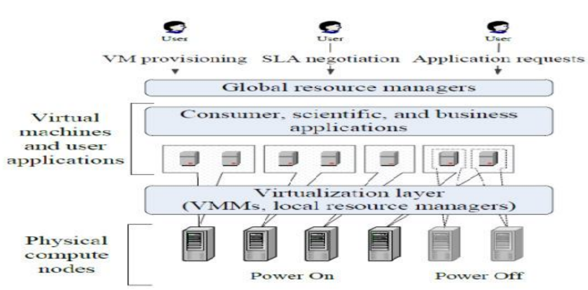

Fig. 1. Cloud System Architecture [16]

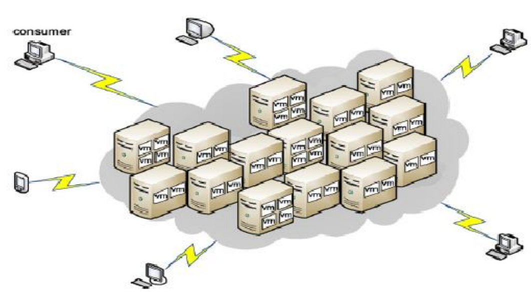

Fig. 2. The cloud Model [3]

Figure 3 shows the energy consumption has risen by 56% by data centers from 2005 to 2010 worldwide. Furthermore, CO_2 emissions of the ICT industries are currently approximated to be 2% of the global emissions. It has been observed that global emissions are equivalent to the emissions of the aviation industries. The energy model is conceptualized on the basis that energy consumption has a linear relationship with processor utilization [15,7]. This means that for a particular task, the processing time of a task and the processor utilization are the required parameters to determine the energy consumption for that task. The utilization Z_i is defined as for a source r_i at any prearranged time.

$$Z_i = \sum_{j=1}^{i} z_{i.j} \tag{1}$$

In (1) it has been observed that t is assigned as the number of tasks running at the current time where as $z_{i.j}$ stands for the resource usage of the task t_j. The energy consumption E_i of a resource r_i at a given time is derived as

$$E_i = (\delta_{\max} - \delta_{\min}) \times z_i + \delta_{\min} \tag{2}$$

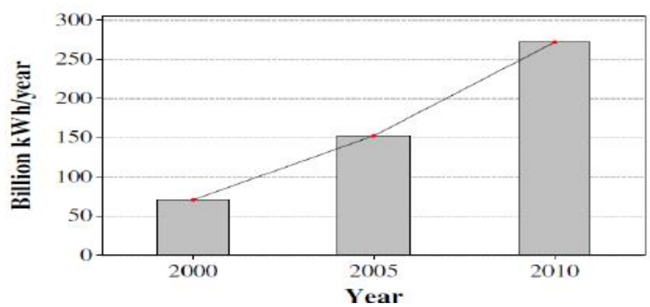

Fig 3. The worldwide data centre energy consumption (2000-2010) [16]

In (2) it has assigned as $\delta_{\max}$ is meaning for the energy consumption at 100% utilization or peak load and $\delta_{\min}$ is assigning the minimum energy consumption as low as 1% consumption or in dynamic mode. In this chapter, it has assumed that the resources in the objective arrangement are comprised of an efficient power saving method in favor of an inactive time slot. Particularly, the energy use of an inactive source at any specified time has been set to 10% of $\delta_{\min}$ .According to the energy consumption, the VMs can be broadly classified into six levels, the idle state and other five levels of CPU utilizations which has shown in Figure 4.

$$E_t(V_i) = \begin{cases} \alpha \quad watts/s, \; if \quad idle \\ \beta + \alpha \quad watts/s, if \; 0\% < CPU \; utilization \le 50\% \\ 2\beta + \alpha \quad watts/s, if \; 50\% < CPU \; utilization \le 70\% \\ 3\beta + \alpha \quad watts/s, if \; 70\% < CPU \; utilization \le 80\% \\ 4\beta + \alpha \quad watts/s, if \; 80\% < CPU \; utilization \le 90\% \\ 5\beta + \alpha \quad watta/s, if \; 90\% < CPU \; utilization \le 100\% \end{cases}$$

Figure 4. Five levels of CPU Utilization [4]

In the present study, the task consolidation algorithm assigns a set M of m tasks to a set N of n cloud resources without violating the time constraints to minimize energy consumption and to maximize resource utilization.

III. PROPOSED ALGORITHM (MINIMIZATION OF IDLE RESOURCE)

Task allocation is a NP-Hard problem in the cloud. Heuristic and meta-heuristic algorithms are the two useful and efficient technologies for scheduling in cloud due to the ability to distribute and deliver the optimized solutions As the idle resources also consume power, the proposed algorithm always minimizes the number of idle resources by allocating a task at an instance to a VM which is currently idle. If no machine is idle it implements ECTC algorithm. The pseudo codes of all the algorithms are as follows:

MaxMaxUtil Algorithm

Input : Task Matrix (mat)
Output : Allocation Table (Alloc)
[minArrivalTime maxArrivalTime]=
sort the tasklist in descending order of required CPU utilization
for each task in tasklist do
find the VM which has currently highest CPU Utilization

ECTC Algorithm

Input : Task Matrix (mat)
Output : Allocation Table (Alloc)
for each task in tasklist do
for each vm in vmlist do
max=-1
E= EnergyConsumedIncludingTheTask(task,vm)
//Allocate the task into the VM to the maximum
energy efficient
if E > max
max=E
allocatedVm =Vm
end if
end for
if allocatedVm !=NULL
allocate task to allocated VM
Update the Alloc table end
if end for
End Algorithm

<u>ECTC with Minimization of Idle VM</u>

Input : Task Matrix (mat)
Output : Allocation Table (Alloc)
for each task in tasklist do
for each Vm in Vmlist do
if CPUUtilization(Vm)==0
allocatedVm=Vm
Return
else
max=-1
E= EnergyConsumedIncludingTheTask(task,Vm)
//Allocate the task into the VM to which it will be
maximum energy efficient
if E > max
max=E
allocatedVm =Vm
end if
end for
if allocatedVm !=NULL
allocate task to allocatedVm
Update the Alloc table
end if
end for
End Algorithm

IV. AN ILLUSTRATION

Consider a set of 10 VMs V={V1,V2,...V10} and a set of 20 independent tasks T={T1,T2,T3,..T20} in which each task Ti has 4 tuples {TaskId, Arrival Time, Processing Time, CPU Utilization}. We have considered the threshold value of CPU utilization as 100%. The task table has been shown in Table 1. The task allocation table for various algorithms such as MaxMaxUtil, ECTC and our proposed algorithm are shown in Tables 2, 3 and 4 respectively.

TABLE 1.Example of Task Table

Task Id	Arrival Time	Processing Time	Utilization
1	1	12	54
2	1	5	62
3	1	7	31
4	1	12	51
5	1	9	67
6	2	8	59
7	2	11	57
8	2	8	31
9	2	10	54
10	2	10	66
11	2	17	61
12	3	17	45
13	3	13	43
14	3	9	59
15	3	7	13
16	3	12	40
17	4	12	63
18	4	11	22
19	4	6	18
20	4	14	33

TABLE 2. TASK ALLOCATION TABLE USING MaxMaxUtil

Task Id	Arrival Time	Processing Time	End Time	Utilization
5	1	1	9	67
2	2	1	5	62
1	3	1	12	54
4	4	1	12	51
3	1	1	7	31
10	5	2	11	66
11	6	2	18	61
6	7	2	9	59
7	8	2	12	57
9	9	2	11	54
8	5	3	9	31
14	10	3	11	59
12	3	3	19	45
13	8	3	15	43
16	7	3	14	40
15	2	6	9	13
17	2	4	17	63
20	6	4	17	33
18	10	4	14	22
19	10	4	9	18

TABLE 3. TASK ALLOCATION TABLE USING ECTC ALGORITHM

Task Id	Arrival Time	Processing Time	End Time	Utilization
1	1	1	12	54
2	2	1	5	62
3	1	1	7	31
4	3	1	12	51
5	4	1	9	67
6	5	2	9	59
7	6	2	12	57
8	3	2	9	31
9	7	2	11	54
10	8	2	11	66

11	9	3	18	61
12	7	3	19	45
13	6	3	15	43
14	10	3	11	59
15	1	3	9	13
16	10	6	14	40
17	2	4	17	63
18	9	4	14	22
19	3	4	9	18
20	8	4	17	33

TABLE 4. TASK ALLOCATION TABLE USING THE PROPOSED MINIMIZATION

OF IDLE VM ALGORITHM

Task Id	Arrival Time	Processing Time	End Time	Utilization
1	1	1	12	54
2	2	1	5	62
3	3	1	7	31
4	4	1	12	51
5	5	1	9	67
6	6	2	9	59
7	7	2	12	57
8	8	2	9	31
9	9	2	11	54
10	10	2	11	66
11	8	3	18	61
12	1	3	19	45
13	4	3	15	43
14	3	3	11	59
15	5	3	9	13
16	7	6	14	40
17	2	4	17	63
18	9	4	14	22
19	5	4	9	18
20	10	4	17	33

VI. SIMULATION RESULTS

The behavior of three task consolidation heuristic with 1000 tasks has been simulated here. Matlab 2012 software has been used for simulation for 1000 tasks. The tasks arrive at the central server queue with a rate of λ having unlimited buffer size. It has taken the arrival interval between the tasks as 1 and the task arrival rate to be 30 in the present studies. The task consolidation algorithms behaviors are demonstrated for 5, 10 and 15 VMs in figures 5, 6 and 7 respectively. The consumption of energy on 15 VMs by varying the task size from 500 to 1500 has been shown in figure 8.

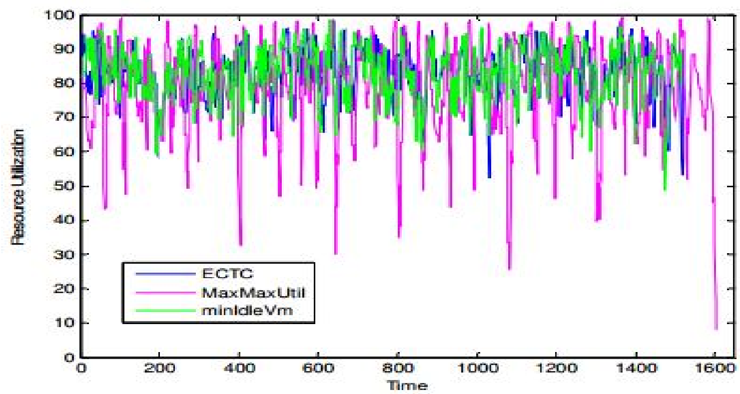

Fig. 5. CPU utilization comparison for 1000 tasks on 5 VMs

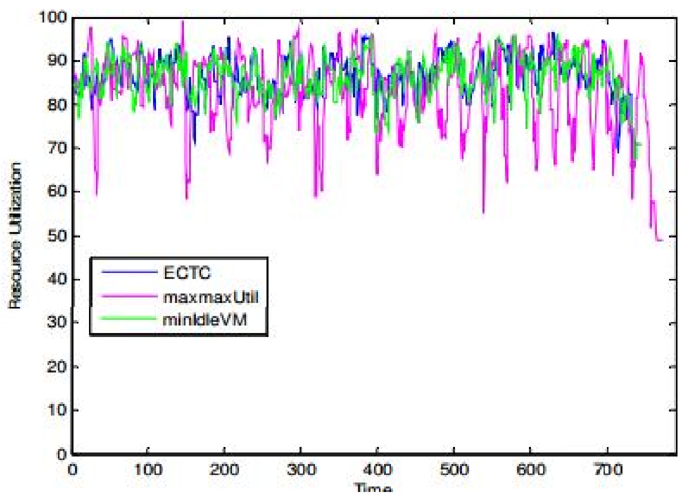

Fig. 6. CPU utilization comparison for 1000 tasks on 10 VMs

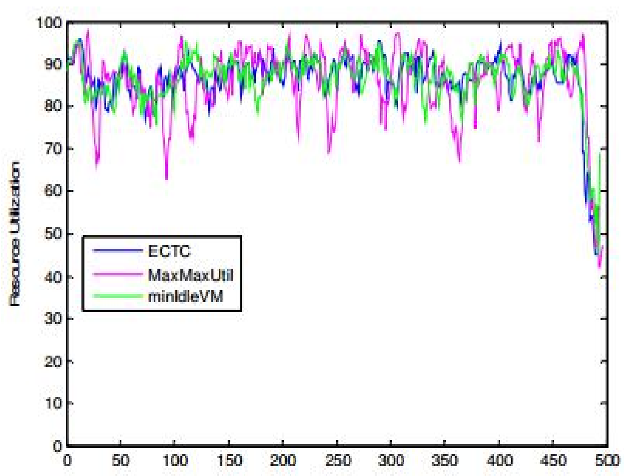

Fig. 7. CPU utilization comparison for 1000 tasks on 15 VMs

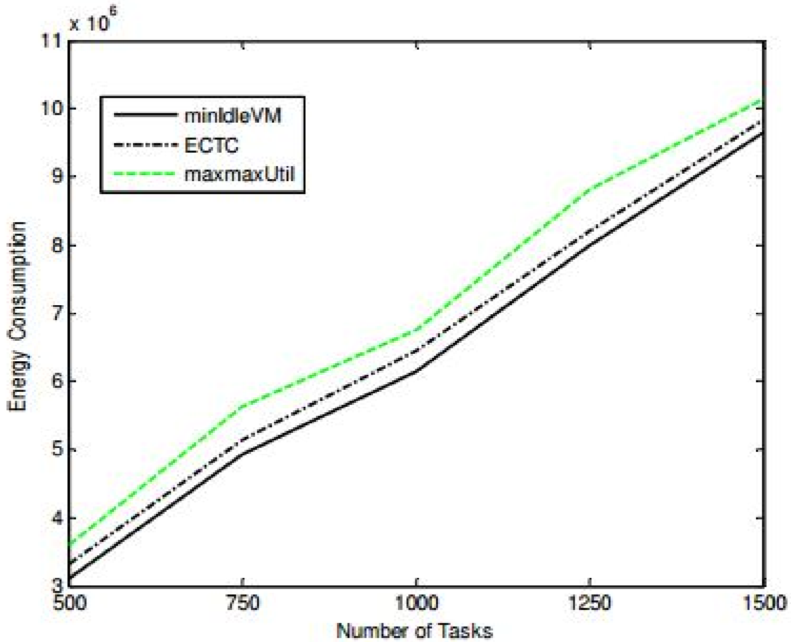

Fig. 8. Energy Consumption for number of tasks on 15 VMs

VII. CONCLUSIONS

The simulation experiments have been successfully carried out which examines the behavior of heuristic task consolidation algorithms. It has also been optimized for energy consumption in a cloud environment. The performance analysis has been demonstrated for the various task consolidation algorithms for the ETC matrix. The results drawn by the shows that the proposed algorithm has saved the energy as compared to the existing algorithms.

REFERENCES

[1] Gaojin,W., Hong,J., Chengzhong, X., Balaji, W., Shengzhong F., Pingchuang,J.,(2011)."Energy-aware hierarchical scheduling of applications in large scale data centers". International Conference on Cloud & Service Computing (CSC) : 158-165

[2] Kim, K. H., Buyya, R., & Kim, J. (2007)" Power Aware Scheduling of Bag-of- Tasks Applications with Deadline Constraints on DVS-enabled Clusters". In CCGrid (7):541-548

[3] Lee, Y. C., Zomaya, A. Y. (2012)." Energy efficient utilization of resources in cloud computing systems". The Journal of Supercomputing, 60(2): 268-280.

[4] Hsu, C. H., Chen, S. C., Lee, C. C., Chang, H. Y., Lai, K. C., Li, K. C., Rong, C.(2011)"Energy-aware task consolidation technique for cloud computing". In Cloud Computing Technology and Science (CloudCom), IEEE Third International Conference on IEEE.: 115-121

[5] Srikantaiah,S., Kansal,A.,Zhao,F(2008)." Energy aware consolidation for cloud computing". In Proceedings of the 2008 conference on Power aware computing and systems, 10: 1-5.

[6] Ali, S., Siegel, H. J., Maheswaran, M., Hensgen, D.(2000)." Task execution time modeling for heterogeneous computing systems". In Heterogeneous Computing Workshop, 2000 Proceedings. 9th . IEEE: 185-199

[7] Fan, X.,Weber,D.,Barroso,L.A.(2007)." Power provisioning for a warehouse-sized computer". In Proc 34th annual International Symposium on Computer Architecture (ISCA ’07): 13–23.

[8] Meisner, D., Gold, B. T., & Wenisch, T. F. (2009, March). " PowerNap: eliminating server idle power. In ACM Sigplan Notices". ACM ,44(3): 205-216

[9] Hsu, C. H., Slagter, K. D., Chen, S. C., & Chung, Y. C. (2014). "Optimizing energy consumption with task consolidation in clouds". Information Sciences,(258): 452-462.

[10] Lee, Young Choon, and Albert Y. Zomaya(2009)." Minimizing energy consumption for precedence-constrained applications using dynamic voltage scaling. & quot; In Cluster Computing and the Grid". CCGRID'09. 9th IEEE/ACM

[11] Tian, Wenhong, Qin Xiong, and Jun Cao(2013)." An online parallel scheduling method with application to energy- efficiency in cloud computing.& quot; The Journal of Supercomputing,(3):1773-1790 International Symposium on..IEEE, 92-99.

[12] Kim, Nakku, Jungwook Cho, Euiseong Seo(2014). Energy-credit scheduler: an energy-aware virtual machine scheduler for cloud systems." Future Generation Computer Systems (32) : 128- 137

[13] Bojanova, Irena, Augustine Samba(2011)."Analysis of cloud computing delivery architecture models." In Advanced Information Networking and Applications (WAINA)". IEEE Workshops of International Conference on IEEE: 453-458.

[14] Beloglazov, A (2013). "Energy-efficient management of virtual machines in data centers for cloud computing. PhD thesis, Department of Computing and Information Systems". The University of Melbourne.

[15] Koomey, Jonathan(2011) Growth in data center electricity use 2005 to 2010. A report by Analytical Press, The New York Times.

[16] Parthasarathi P., Jayram P., (2016) Minimizing Cost using Cloud Computing with RFID Based Supply Chain Management **IJ C T A**, 9(10), 2016, pp. 453-461.