International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 5 – No.1,January 2013 – www.ijais.org

Energy-Efficient Job Scheduling and Allocation Scheme for Virtual Machines in Private Clouds

Shailesh S. Deore
Research Scholar
JJT University, Jhunjhunu
Rajasthan, India.

Ashok Narayan Patil, PhD.
Principal, Vasantrao Naik Arts and
Science,College,Shahada.

ABSTRACT

Cloud Computing is a promising paradigm in deployment of software and there survive. Computer researcher says a magnificent sentence, cloud computing is next generation operating system. Technologies such as cluster, grid, and now, cloud computing, have all aimed at allowing entrance to large amounts of computing power in a fully virtualized, para-virtulized manner. The Energy efficiency of information and communication technology becomes more and more vital due to elevate of energy costs and the world wide desire to decrease CO2 emissions [13],[14]. In this paper we argues energy-efficient job scheduling and allocation scheme that minimize number of hosts, so amount of energy conserve.

General Terms

Cloud Computing, Green Computing, Scheduler.

Keywords

Energy, Virtual, energy efficient, Workload, VM, VM request

1. INTRODUCTION

Energy efficiency of information and communication technology has become an important paradigm in companies and public administration [13]. While hardware cost is trying to decrease in one hand, cost of power is increase in other hand. In addition of that cooling these servers in labs, office enjoinments large amount of CO2 emission which is wide effects of green computing [1], [13].In previous study conserve energy by provide algorithmic approach, in this paper we propose same scheme but job allocation strategy is different so minimize number of hosts which runs number of guest.

Once hosts outside, inside of data centers in private cloud environments contribute signicantly to overall IT energy utilization, consumption so we are concentrating on Hosts which runs number of virtual machines. A variety of scheduling algorithms have been developed in previous years. FCFS, Greedy, round robin, Priority Scheduling, Shortest Job First, Back Filling in operating system for scheduling number of processes coming from users. All this algorithms are to save less amount of energy. This paper is focus on minimize the number of hosts, lease from consumer schedule and allocate to virtual machine such that less amount of Hosts required so maximum amount of energy conserve.

2. REVIEW OF RELATED WORK

This work review and discuss many approach viz. algorithm, methods, paradigms, techniques, how to schedule and allocate virtual machines running on physical machines call Hosts and also concentrate on energy consumption less, optimization, fully workload distribution, exploitation with physical machine rate ability.

Jiandun Li et al [2] introduce a hybrid energy-efficient scheduling algorithm for private clouds, concentrated on load balancing, load migration of virtual machines, measure response time, if response time is directly propositional to energy, so they minimized response time in that algorithm, design states of virtual machines and observe there characteristics with energy consumption and conservation.

Saurabh Kumar Garg et al [3] presented a few scheduling techniques that could utilize heterogeneity across various data centers.

Akshat Verma et al [4] also present a dynamic power aware strategy that scale voltage, frequency for save energy.

Gregor Von Laszewski et al [5] proposed scheduling of virtual machine in a compute cluster to reduce power consumption through Dynamic Voltage Frequency Scaling (DVFS), implementation of energy efficient algorithm to allocate virtual machine that work on minimum voltage, frequency to save more power.

Aman Kansal et al [6] states virtual machine power metering and provisioning architecture i.e. Joulemeter measure power of virtual machines per second in watt.

T.Tamir [7] discuss scheduling bully selfish jobs precedence-constraints, i ≺ j means that job j cannot start being processed before job i is completed. This article author was considering selfish bully jobs who do not let other jobs start their processing if they are around. Officially, author define the selfish precedence-constraint where i ≺ s j means that j cannot start being processed if I has not started its processing yet.

Pradeep kumar Sharma et al [8] discuss the algorithms for creating the small cloud using simulator CloudSim, and some key feature of conserving the energy in cloud with the help of migration of virtual machines in between data centers. The redundant datacenter consumes the large amount of energy which becomes the challenging for the data centre.

3. BACKGROUND

In previous study [9],[10] energy-efficient scheduling scheme is same but only job allocation strategy to virtual machine is easy-backfilling with first come first serve [11] , if saw a keen observation on hosts i.e. physical machine Table.1, we realize that if minimum number of host is always significant for conserve energy.

Table1. Measurement of Energy (J) and Power (W)

Physical Machine / Hosts H	Time in S	Energy in J	Power in W/S
Single Host Power ON	60	4236	70.6
Single Host Power OFF	60	5364	89.4
Single Host Sleep/standby	60	552	9.2
Laptop Power ON	60	2592	43.2
Laptop Power OFF	60	3078	51.3
Laptop Sleep/Standby	60	228	3.8

In Previous EESS with FCFS [9],[10] scheduler allocate number of VM to VM request using simple First Come First Serve (FCFS) but if think about to conserve energy on physical machines then minimize number of host H i.e. Physical machines. If we backfills number of VM on Host and try to avoid start new Host for allocation of new VM to consumer, because of we know maximum amount of energy required for starting Host, again if we use backfill with First come first serve then waiting time is also reduced so consumer fill happy about our scheduling, previous study consumer waits long time for VM and it fills misery.

Basic architecture of private cloud is such that each Host contain four VM, Host H_1 contain EESS , see fallowing fig.1 contain

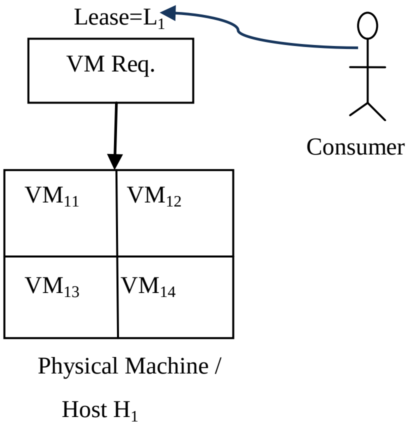

Fig.1: Host and Guests in Private Cloud

3.1 EESS with FCFS

This approach number of Hosts required three and waiting time is increased when scheduling VM to VM requests, consumer who provide their requests is fills misery about VM allocation time, as shown in Fig.2, Suppose consumer submit 4 leases, such that L_1={3 }, L_2={3}, L_3={1}, L_4={1} , we have total 8 VM request then EESS with FCFS allocate VM request as fallowing Fig.1

3.2 EESAS with EASY Backfilling FCFS

In this approach [11],[14],[15] number of Hosts required two, waiting time reduces , L_3 Backfill with L_2 so consumer fills happy as compare to previous approach and waiting time of lease L_3 reduces, number of hosts required two and conserve more energy as shown in fig.2.

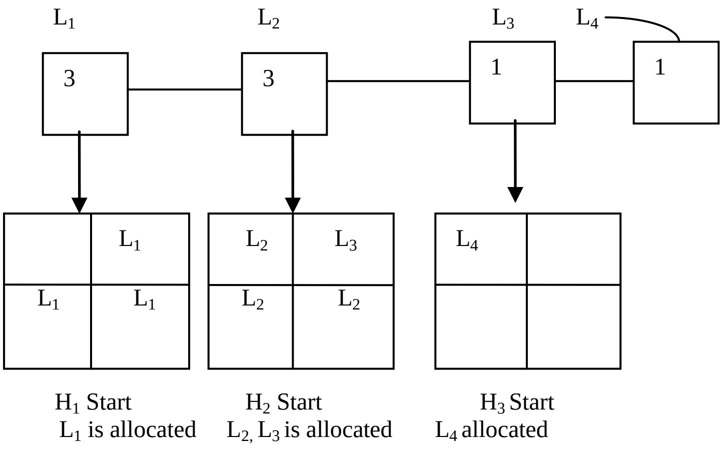

Fig.2: EESC with FCFS job allocation to VM

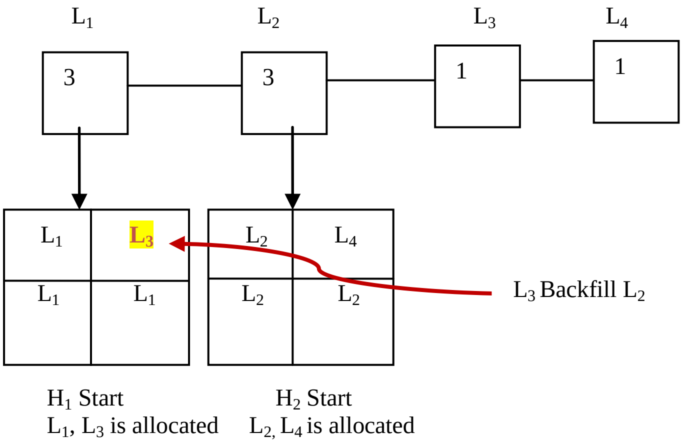

Fig.3: EESC with, Easy Backfill with FCFS

4. ENERGY-EFFICIENT SCHEDULING AND ALLOCATION SCHEME (EESAS)

In this section, we discuss EESS with Easy Backfill FCFS all parameters is same only allocation strategy is changed, at time of scheduling EES scheme use migration, clone, pause, resume capability of virtual machines to conserve more energy. Consumer write request of VM using lease management system call LMS, these number of leases contain number of virtual machine request EESS schedules all VM request to VM using EESS policy.

4.1 Pseudo Code of EESAS

Steps of Algorithm

```
// Set default values of U, H, $VM_{Hi}$, i , j , J ,L, $VM_S$ , $VM_T$ ,
VM .

1. Start
2 .If (j <= 0)  //VM request equals to zero or less than   zero
3. Exit();
```

```
4.end if .
5. If ( j < J)     // VM request is less that Total number of VM
                    // request  i.e. least load first
6. PowerONVM ( j, H, $VM_{Hi}$ );  // power on Virtual machines
7  if ($VM_{Hi}$ == Pause) then Resume($VM_{Hi}$);
8. if (L==P) then Clone(VM)
9. end if.
10.  $VM_{Hi}$ ← j // Assign the number of jobs to Virtual
                 //machines by using EASY Backfill with First
                 //come first serve
11. else
12. if ( j >= J )    // VM request is greater  than or equals  to
    //total number of VM request , maximum load schedule
    // using migration
13. Find  ( $VM_S$ , $VM_T$ );      // required Migration so find
    source and Target Virtual machine
14. $VM_T$ ← PowerOFF  $VM_{Hi}$  state and on its Teleport-In
    if ($VM_T$== Running) then Pause($VM_T$);
15. $VM_T$ ← Teleport-In($VM_T$)  //TeleportIn state in Target
//virtual machine
16. $VM_S$  ←  Active $VM_{Hi}$ state as a source
17. Migration ($VM_S$, $VM_T$);
18.end if
19. end if
20. Update ($VM_{Hi}$ , j, J, L);
21. end of EESS.
```

Function explanation use in EESS

Pseudo code of PowerONVM(j,H, VM_{Hi})

Set of steps

```
// parameters required
// J ← incoming jobs from users  , H ← Total no of hosts  ,
$VM_{Hi}$ ← set of virtual machines to start for jobs, stepping
variables  n , i , m
```

```
1.   $VM_{Hi}$  ← j       // value of j transfer to $VM_{Hi}$
2.    m = 0;
3.   While (m < =$VM_{Hi}$)
       for (n=1;n<=H; n++) // number of Host
        for(i=1; i<=VM; i++) // virtual machine on that   //
                                        particular Host
         Start virtual machines
         m ++;
     end for loop.
     end for loop
4.   end of while loop
5.   end.
```

Pseudo code of Migration ($VM_{S,}$ VM_T)

Set of steps

```
// parameters required
// $VM_{s,}$ $VM_T$
```

```
1.  $VM_s$ = Active state, Running state
2. $VM_T$=Teleport-In state
3. Apply ($VM_s$ → $VM_T$ )
4. end.
```

EESAS is implement on base of three conditions i) No workload i.e zero workload ii) Minimum workload and, iii) Maximum workload , when minimum workload then start virtual machine as per VM request, if no workload then do not start any VM for conserving energy purpose, maximum workload then apply migration of VM , clone one VM to multiple VM , in this EESAS with easy backfilling FCFS only allocation of job to VM is change.

5. EXPERIMENTAL RESULTS

We present series of experiments to exemplify the outcome of EESS discussed in previous section 4. The test bed is composed on 4 personal computers (HCL 3.06) each have LX INFINITI PRO BL 1205 CPU P4 524@3.06GH with 2GB memory connected to 10/100 Mbps switched Ethernet. Each host contain 4 VM, host1 acts as a scheduler, calculated conserve energy (E) in Joule(J) using Joulemeter1.2 [6], [16] , total time (T), power (W) , number of VM N, number of leases L, VM request from consumer J. Using VirtualBox3.1 [13] for creating private cloud environment, APIs start, stop, pause, resume, clone, and migrate virtual machines in JDK1.7.0 for algorithmic implementation of EESS. Consider EESS with FCFS and EESAS with Easy backfilling FCFS and show the comparisons which is better for energy saving paradigm.

5.1 Comparison

Energy Conserved (E): In Fig.4 we are cleared demonstrated that EESAS with Easy Backfill FCFS is more energy conserved as compare to EESS with FCFS, but if elaborate private cloud environment then EASY Backfill with FCFS allocation strategy is always better than previous approach.

Number of Physical Machines (H): EESAS with Easy Backfill FCFS is better than EESS with FCFS as shown in fig.5, because of number of hosts H is less required and waiting time is reduces and consumer fills happy as compare to EESS with FCFS allocation strategy through energy saving criteria. Virtual machines required both allocation strategy is same number but waiting time is reduced so some amount of energy is conserved through Easy Backfilling.

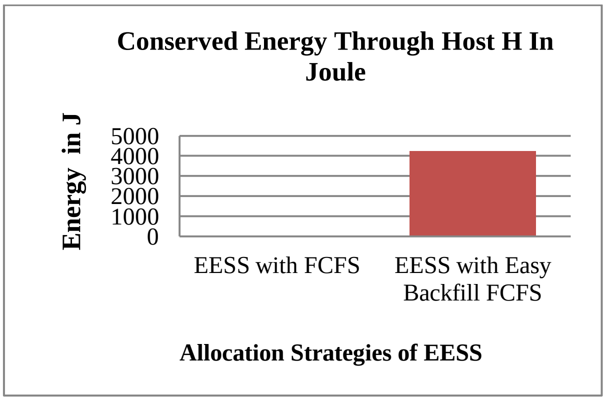

Fig. 4: Conserved Energy J of EESS through Hosts

Table 2. Detailed Data

Metrics / Methods	Total Energy in J by H	Total Time in S	Power in W per min.	Number of VM required as per Method N	VM request from consumer	Leases(L)	Energy per Minute in J	Average Waiting Time T	Number Hosts H
EESS with FCFS	912420	1800	506.4	8	8	4	30414	15.14	3
EESAS with EASY Backfilling FCFS	738922.8	1624	436.2	8	8	4	26172	13.47	2

Table 3. Conserved Energy (E) in Joule

Energy (E) In Joule / Methods	Conserved Energy through Host per minute	Conserved Energy through Host per Total time
EESS with FCFS	0	0
EESAS with EASY Backfilling FCFS	4242	173497.2

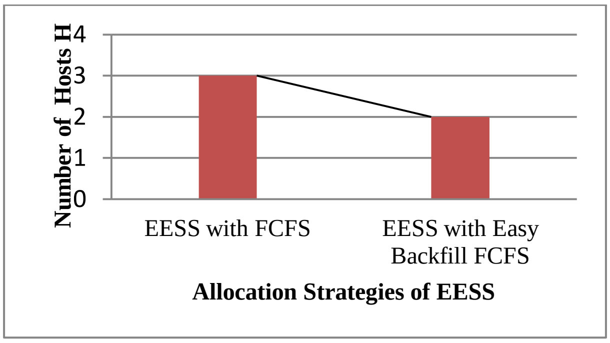

Fig. 5: Number of Hosts as per Allocation Strategies

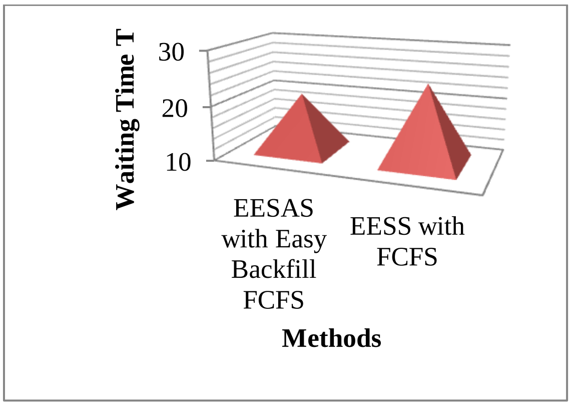

Fig. 6: Waiting Time as per Allocation Strategy

Waiting Time: As shown in fig.6 waiting time of EASY Backfill FCFS, VM to VM request is less as compare to EESS with FCFS, so consumer fills happy and virtual machine is start but VM request is not allocated, VM wait for job so amount of energy is required their stand by phase, to conserve that energy, apply Easy backfill with FCFS.

6. CONCLUSION

Cloud computing is most important trend in data center in private cloud environment. One of strongest credit is “green” alternative offered to consumer. EESAS with Easy Backfilling FCFS represents attractive commercial scheme to reduce energy consumption at the consumer side which is really need in data centers. Cloud providers need of scheduling schemes not only conserving more energy and consumer satisfaction in service providing.

This paper represent scheduling scheme that reduce energy consumption as well as consumer satisfaction also providing to reduce their waiting time for virtual machine at highest priority. From this long analysis we conclude that there still exist some gaps that must be covered, Green Computing is help to reduce that gaps so we need to move towards Green computing.

7. REFERENCES

[1] Andreas Berl, Hermann de Meer, “A virtualized energy-efficient office environments”, e-Energy 10, 2010, Apr 13-15, pp.11-20.

[2] Jiandun Li, Junjie Peng, Wu Zhang,”A Scheduling Algorithm for Private Clouds”, Journal of Convergence Information Technology,2011, Volume 6, Number 7,pp. 1-9.

[3] Saurabh Kumar Garg, Chee Shin Yeo, Arun Anandasivam, Rajkumar Buyya, “Environment-conscious scheduling of HPC applications on distributed cloud-oriented data centers ”, Journal of Parallel and Distributed Computing, 2011,vol.71, no.6, 732-749.

[4] Akshat Verma, Puneet Ahuja, Anindya Neogi, “Power-aware dynamic placement of HPC applications”,2008, Proceedings of the 22nd International Conference on Supercomputing (ICS’08),Island of Kos, pp.175-184.

[5] Gregor von Laszewski, LizheWang, Andrew J. Younge ,Xi He ,”Power-Aware Scheduling of Virtual Machines in DVFS-enabled clusters”, cluster 09 IEEE international on Cluster , 2009, pp.1-11.

[6] Aman kansal, Feng Zhao, jie Liu, Nupur Kothari, Arka A. Bhattacharya , “Virtual machine power metering and provisioning”, copyright 2010 ACM, 2010.

[7] Tami Tamir, “Scheduling with bully selfish jobs”, Theory of Comput Syst, vol.50, no.1, 2012, pp.124-146.

[8] Pradeep Kumar Sharma, Chandana Das,” Energy efficient scheduling in cloud computing ”, Inventi Impact: Cloud Computing,2012.

[9] Shailesh Deore, A.N.Patil, Ruchira Bhargava,” Systematic Review of energy efficient scheduling techniques in cloud computing” International journal of computer application,Vol.52, 2012, Number 15.

[10] Shailesh Deore, A.N.Patil, Ruchira Bhargava,”Energy Efficient Scheduling Scheme for virtual machines in cloud computing” International journal of computer application, Vol.56, 2012, Number 10.

[11] Juan Manuel Ramrez-Alcaraz, Andrei Tchernykh, Ramin Yahyapour, Uwe Schwiegelshohn, Ariel Quezada-Pina, Jose Luis Gonzalez-Garca, and Adan Hirales-Carbajal, “Job allocation strategies with user run time estimates for online scheduling in hierarchical grids”, J. Grid Comput., ,2012, October 10,pp.95-116.

[12] Virtual Box available on www.virtualbox.org

[13] Ismael Solis Moreno, Jie Xu, “Energy-Efficiency in cloud Computing Environments : Towards Energy Saving without Performance degradation”,In International Journal of Cloud appliocation and Computing, Vol.1, Issue.1, 2011, pp.17-33.

[14] R.Buyya, Beloglazov A., Abwajy J.,”Energy-Effcient Mangement of Data Center Resources for Cloud Computing: A vision, Architectural elements, and open Challenges ” In proc. Of 2010 International conference on parallel and Distributed Processing Techniques and Application, las Vegas, NV, USA.

[15] Francesc Guim, Ivan Rodero, Julita Corbalan, A. Goyeneche, “The Grid Backfilling:A Multi-site data mining architecture with data mining prediction techniques”,2008, Grid middleware and services,published by Spinger US, pp.137-152

[16] Joulemeter1.2 available on: http://research.microsoft.com/en-us/default.aspx

[17] I.Foster, “The grid: Computing without bounds ” , In scientific Journal , Vol.288, No. 4, pp.78-85.

[18] D. Tsafrir, Y. Etsion, and D. G. Feitelson,” Backfilling using system-generated predictions rather than user runtime estimates” In the IEEE TPDS, 2006.

[19] Y. Zhang, W. Sun, and Y. Inoguchi,” Cpu load predictions on the computational grid”, Cluster and Grid computing, 2006.

[20] R. V. Patil, K. C. Jondhale 2010 , ” Edge based technique to estimate number of clusters in k-means color image segmentation”, IEEE Internatinal conference on Computer Sciene and Information Technology, Chengdu , China, vol. 2, pp.117-121.

[21] P. S. Patil, S. R. Kolhe, R. V. Patil, P. M. Patil , “Performance Evaluation in Iris Recognition and CBIR System based on phase congruency”, International Journal of Computer Applications, Vol.47(14),2012, pp.13-18.