Energy-Efficient Scheduling Scheme for Virtual Machines in Cloud Computing

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

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

ABSTRACT

A Cloud Computing is subscription-based service where we obtain networked storage space and computer resources. Technologies such as cluster, grid, and now cloud computing are decide allowing access large amounts computing power in fully, para-virtulized manner by amassed resources i.e. virtual machines to consumer. Cloud computing denotes energy efficiency in all components of computing systems that is hardware, software, local area network. Energy computing has to achieve manifold objectives of energy consumption and utilization improvement for computing paradigm that are not pay-per-use such as cluster and grid revenue maximization as another metric for cloud computing architecture. Microsoft Corporation provides Joule meter tool to measuring energy of computing software devices, for researcher to observe energy conservation and consumption [10],[14]. This paper we proposed Energy-Efficient Scheduling Scheme call EESS for virtual machines that distribute maximum workload on minimum number of virtual machine's so less amount of energy consume, to test the scheme cloud environment created using VirtualBox.

General Terms

Cloud Computing, Scheduler,

Keywords

Virtual machine request, virtual machine, energy, Workload, EESS,VM, VM request, power.

1. INTRODUCTION

Now a day's cloud computing is a promising computing technology which is to support virtualization, scalable resource utilization and provide services such as IaaS, SaaS, PaaS. A Computer scientist predicting that cloud system is next generation operating system. Google engineers say a magnificent sentence maintains thousands of servers, warned that if power consumption continues to increase, power cost can easily overtake hardware cost by a large margin [1].

Cloud computing is open source for cloud developers. The development of hypervisors Xen, KVM [17], VMware ESXi [25], VirtualBox, Eucalyptus are triggering development of commercial and open source cloud environment. It can offer services on base of energy, power pay-per use model. Self-service, per usage metering, billing, elasticity, customization these are desired features of cloud [20],[23],[24]. There are different types of cloud that can subscribe by consumer depending upon needs such as public cloud, private cloud, community cloud, hybrid cloud. Let us consider a simple example in real life, when we plugging an electric instrument into electric outlet, we neither think how electric power generated, where generated, how it is gets to outlet this is possible because of electricity is virtualized. It is readily available from a wall socket that hides power generation stations, cloud computing is also same as for virtualization of computer resources to consumer pay providers based on their usage that is call pay-per-use model. So we say a magnificent sentence about cloud, Cloud Computing is a Nutshell [20],[21],[22].

A Scheduler is required to schedule number of virtual machine as per virtual machine request from consumer provide in form of lease. Scheduler schedules number of virtual machine request such that to conserve maximum energy and achieve greater degree of load balancing and less resource utilization. It is a challenging problem to effectively schedule virtual machine request in form of lease to virtual machine. In this paper we discuss review of related work, background of our Energy efficient scheduling scheme (EESS), detail of algorithmic design and experimental result, conclusion this paper. We refer virtual machine as VM and virtual machine request as VM request in this realm.

2. RELATED WORK

In this decade we discuss many approaches viz. algorithm, methods, paradigms, techniques, how to schedule virtual machines running on physical machines and also concentrate on energy consumption less, optimization, fully workload distribution, exploitation with physical machine ratability.

Jiandun Li et al [2] introduce a energy-efficient scheduling approach for private clouds to reduce coming request response time, balance workload when data centre is running in low power mode and design algorithm on the base of pre power techniques and Least-load first algorithm, experimental result show save more energy, archive higher level of load balancing.

Jiandun Li et al [3] introduce a hybrid energy-efficient scheduling algorithm for private clouds, concentrated on load balancing, load migration on the base of state of virtual machines, measure response time, if response time is directly propositional to energy. So they minimised response time in that algorithm, design states of virtual machines and observe there characteristics with energy consumption.

Saurabh Kumar Garg et al [4] presented a few scheduling polices that could utilize heterogeneity across various data centres. Akshat Verma et al. [5] also present a dynamic power aware strategy to save energy.

Chuliang Weng, Zhigang Wang et al [6] present hybrid scheduling framework for the CPU scheduling in the virtual machine monitor. Two types of applications are high-throughput type and concurrent type. Virtual machine sets concurrent type when majority of workload is concurrent applications in order to reduce cost of synchronization. Otherwise it is set to high-throughput type by default. Experiments and results show that framework and scheduling strategy is flexible to improve performance of virtual machine.

Gaurav Dhiman, Giacomo Marchetti et al [7] focus on vGreen: multi-tiered software system for energy efficient computing in virtualized environments, developed MPC balance algorithm that concentrates on CPU and memory utilization decrease, amount of power save up to 15% to 20% of total power required. Power is directly propositional to energy so amount of energy also save.

Gregor Von Laszewski et al [8] proposed scheduling 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.

Bo Li, Jianxin Li et al [9] states Energy aware heuristic algorithm on base of distributes workload in virtual machine with minimum number of virtual machines or nodes required that workload. So that workload migration, workload resizes virtual machine migration these approaches are used in algorithm.

Aman Kansal et al [10] states virtual machine power metering and provisioning architecture i.e. Joulemeter measure power of virtual machines per second in watt.

T.Tamir [11] proposed scheduling bully selfish jobs precedence-constraints, $i \prec j$ means that job j cannot start being processed before job i is completed. In this paper author consider selfish bully jobs who do not let other jobs start their processing if they are around. Officially, author define the selfish precedence-constraint where $i \prec_s j$ means that j cannot start being processed if I has not started its processing yet.

Pradeep kumar Sharma et al [12] proposed 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 center.

3 BACKGROUNDS

Energy saving paradigm is a complex activity compare to other paradigm, as practical implementation of that methods difficult, various tools required, again that tool have capability support virtualization, para-virtualization for maximum workload balancing on virtual machines provides to consumer.

3.1 Basic Architecture of Cloud Computing

In basic architecture of cloud computing, hardware virtualization allows running multiple operating systems and software stack on a single physical platform [20], as depicted Fig.1 the virtual machine monitor (VMM) call hypervisor act as a interface between hardware and virtual machines. In our realm VirtualBox is a hypervisor which control and handles number of virtual machines, scheduler is also design in hypervisor so it schedules virtual machines on different host also, hypervisor have capability migrate, clone, pause, resume virtual machines on different physical machines through application programming interface (API). Consumer provide virtual machine request in form of leases which provides by clouds providers on web, that lease contain configuration of virtual machines, application type (SaaS, PaaS), priority of lease, time and consumer personal information for payment detail such as pay-per-use model.

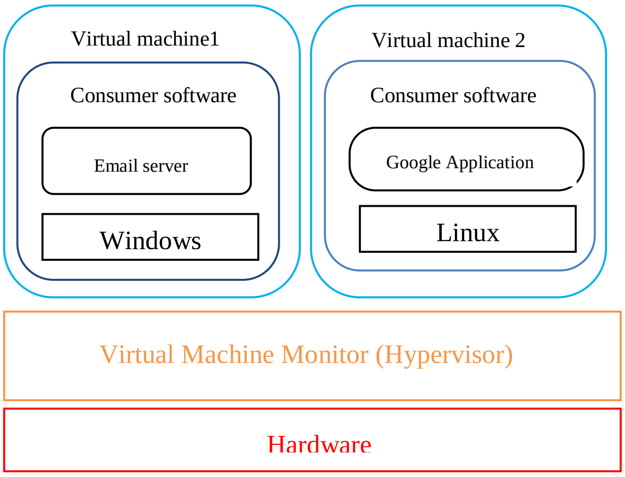

Fig 1. Basic Architecture of Cloud Computing

3.2 VirtualBox

Sun-Oracle provide open source virtualBox3.1 [13] support virtualization, para-virtualization, Xen support para-virtulization, DVF simulator run processor on low frequency, voltage so amount of energy conserve. We go through algorithmic implementation through API provide by VirtualBox, basic concept is a host that is physical machine run number of guests that is virtual machines as we previous discuss in Fig.1, through API we measure and observe energy consumption of each virtual machines run in different states. Fig. 2 shows measurement of energy in joule and power in Watt per second using joulemeter1.2 [14].

Table 1. Measurement of Energy (J) and Power (W)

States of Virtual machine (VM)	Time in S	Energy in J	Power in W
4 VM running at a time	60	1464	24.4
3 VM running at a time	60	972	16.2
2 VM running at a time	60	552	9.2
1 VM running at a time	60	420	7
0 VM running at a time	60	0	0
1 VM powerdown	60	564	9.4
2 VM powerdown at a time	60	1704	28.4
3 VM powerdown at a time	60	2538	42.3
4 VM powerdown at a time	60	3408	56.8
1 VM pause	60	84	1.4
1 VM resume	60	114	1.9
1 VM Teleport-In state	60	378	6.3
1 VM PowerON	60	1848	30.8

A keen observation of energy measurement in virtual machine we conclude that avoid PowerOFF, PowerON of virtual machines, apply migration, clone, pause, resume to save energy. VirtualBox3.1 support all these features, using peak migration of Virtual machine's from one host to another for workload distribution, single virtual machine on running state but do not perform job i.e. VM request required energy 420 J, more than one VM running at a time energy required in terms of quadruple, virtual machine PoweON acquired maximum energy approximately 1848J, VM PowerOFF energy required 564 J, VM in pause/resume state energy required 0J.

In this scenario, we say that running VM but not perform job or completion of job move that particular VM into pause state, VM powerdown avoid instead move into pause state , if required then again resume state to save energy.

A single VM_1 perform more than one VM request & find VM_2 perform no job then migrate VM_1 to VM_2, purpose of our approach is maximum use migration of virtual machines save more energy and maximum workload distribution.VirtualBox3.1 provide Teleport() function through API, select source VM with minimum work load act as a source and find VM with subminimum workload call target VM. Teleport source VM to Target VM. Migration is two type i)static migration ii)dynamic migration that also depend upon lease which is demand by consumer and how our algorithm is schedule workload. In our approach is run application specific policy basically three types of policy a]general-purpose policy b]Architecture-specific policy c]application-specific policy .

3.3 Joulemeter

Microsoft Co-operation provides a glorious software tool call Joulemeter1.2 [14] with manifold objectives; help cloud developer easily measure power usage of VM, server, desktops, laptops and individual software applications running on Host. Joulemeter1.2 provides power consume by CPU, monitor, specific application total power required by system in watt per timestamp in millisecond [10].

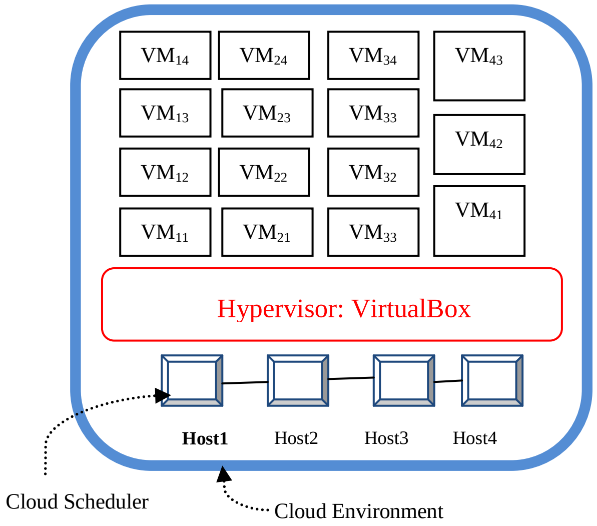

Fig 2. Combination of Host and Guests using VirtualBox

E=Energy in Joule , P=Power in Watt, T= Time in second and N=number of VM, so we can write fallowing expression

$$E, P \propto N \propto T \qquad (1)$$

so we convert power into energy using physics conversation [15]

$$E_J = P_W \times T_S \qquad (2)$$

Combine architecture of Hosts and number of Guests that we implement as shown in Fig.2, it contain total 15 VM, 4 Host and VirtualBox hypervisor, scheduler is design on Host1 that schedules all virtual machines using energy-efficient scheduling scheme(EESS), when we implement EESS then it easily handle all virtual machines on different Hosts. In our approach there is no limitation for Host and Guest, Hosts are connected to each other using LAN, IWebsessionManager interface is use by VirtualBox to handle virtual machines on different Hosts through IP address.

4. ENERGY-EFFICIENT SCHEDULING SCHEME (EESS)

In this scenario we introduce energy-efficient scheduling scheme call EESS with migration, clone, pause, resume basic concept is introduce using minimum load distribution, first come first serve [16], hybrid energy efficient scheduling algorithm [2],[3] incoming VM request call jobs minimum then start virtual machines if request is increased beyond our capacity then apply migration for workload distribution. If VM request is platform, software then applies cloning of VM. Consumer provide request in lease's using lease management system (LMS) collect these leases in queue, virtual machines are also in queue, EESS schedules VM request to virtual machine, implementation is as shown in below fig.3 working procedure of EESS.

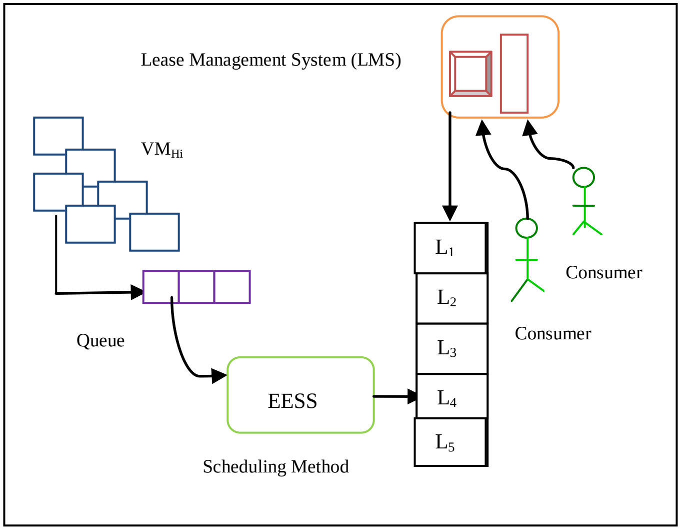

Fig 3. Working procedure of EESS

Consumer provide lease L to lease management system(LMS), incoming jobs collect in set of lease's i.e. $L_{1,}$ $L_{2,}L_{3,}$……$L_{n.}$, algorithm serve these jobs to virtual machines as their configuration and requirement such as

Job_1=VM_{11}

$Job_{2=}VM_{12}$

$Job_{3=}VM_{13}$

. .

. .

$Job_{n=}VM_{1n}$ (3)

4.1 Characteristics of EESS

- Workload distribute on less number of virtual machines.
- Use migration, pause, resume, clone of virtual machines when workload distribute.
- Scheduling of virtual machines to number of jobs less deployment time.
- Response time long of any virtual machine then it is beyond for consideration or not consider.
- Utilization of workload on less number of virtual machine.

- Do not restart shutdown virtual machine again and again.
- Virtual machine running state but not performing any Virtual machine request then change into pause state, again if required then change into resume state.

EESS is design on the base of above characteristics for save energy number of virtual machines, total time, full workload utilization on less number virtual machines, these are important metrics of our scenario.

4.2 States of Virtual machine

Virtual machine is running on different modes, we say state of virtual machine categories in different states and use these states in our EESS see fallowing states.

- **Running:**-virtual machine is ON and waits for VM request.
- **Active:**-Virtual machine is running and performs VM request or job.
- **PowerON:**-Virtual machine is just powering on state
- **PowerOFF**:-Virtual machine is shutting down state.
- **Pause:**-Virtual machine is running state to pause state.
- **Resume:**-Virtual machine is pause state to resume (running) state .
- **Teleport-In:**-Target Virtual machine wait for sourceVM .

On the base of these states we state EESS basically Running, Pause, Teleport-In state required again & again, see fallowing points to be seriously consider that is PowerON virtual machine maximum energy required, PowerOFF virtual machine amount of energy required so it's better rather that PowerON, PowerOFF use Pause, Resume states of virtual machine to save amount of energy that describe in Table1.

4.3 Parameters use in EESS

U - set of Hosts (physical machine) and Guests (Virtual machine) ,U= {H_1,H_2,H_3,….n,VM_1,VM_2,VM_3…. VM_n}

H -Total no of Host's H ∈ U , H=1, 2,3,4…. n

VM_{Hi} - set of virtual machine on each each Host H and i=1,2,3,4, ……n, i for virtual machine, VM_{Hi} ∈ U .

L= number of leases, L_1, L_2, L_3 …... L_n

P= Lease contain VM request such as consumer required Platform as a Service.

j- Incoming VM requests from consumer

J- Total number of VM requests from consumer

VM_{Hi} =

{

Host=1 , i=1,2,3,4…n , VM_{11},VM_{12},VM_{13},VM_{14}.VM_n

Host=2 , i=1,2,3,4…n , VM_{21},VM_{22},VM_{23}, VM_{24}.. VM_n

Host=3, i= i=1,2,3,4 , VM_{31},VM_{32},VM_{33}, VM_{34}.. VM_n

Host=4 , i= i=1,2,3,4 , VM_{41},VM_{42},VM_{43}, VM_{44}.. VM_n

}

VM - Total number of virtual machine in each host (VM=1, 2,3,4 ……n)

VM_S - Virtual machine source required for migration (always in Active state before migration)

VM_T - Virtual machine target required for migration (always in Teleport-In state before migration)

4.4 Function use in EESS

PowerONVM (j, H , VM_{Hi}) :- Virtual Machines start such as equals to number of incoming VM request j.

Migration (VM_S , VM_T) :- Migrate virtual machine from VM_S to VM_T

Pause (VM) :- Pause the running Virtual machine VM for energy save.

Resume (VM) :- Resume the pause Virtual machine VM for energy save .

Clone(VM):- Cloning of one VM to another VM on same H .

Update (VM_{Hi} , j, J, L):- update all values of VM_{Hi} , j, J, L.

Find (VM_s,VM_T):- finding source VM and destination VM.

Exit() :- end of algorithm.

4.4 Pseudo Code of EESS

Steps of Algorithm

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

Start
If (j <= 0) //VM request equals to zero or less than zero
    Exit();
end if .
If ( j < J)   // VM request is less that Total number of VM
              // request i.e. least load first
    PowerONVM ( j, H, $VM_{Hi}$ ); // power on Virtual machines
    if ($VM_{Hi}$ == Pause) then Resume($VM_{Hi}$);
    if (L==P) then Clone(VM);
    end if.
    $VM_{Hi}$ ← j // Assign the number of jobs to Virtual
                  //machines by using First come first serve
else
    if ( j => J )  // VM request is greater than or equals to
                   //total number of VM request , maximum load schedule
                   // using migration
        Find ( $VM_S$ , $VM_T$ );    // required Migration so find
                                      source and Target Virtual machine
        $VM_T$ ← PowerOFF $VM_{Hi}$ state and on its Teleport-In
            if ($VM_T$== Running) then Pause($VM_T$);
        $VM_T$ ← Teleport-In($VM_T$) //TeleportIn state in Target
        //virtual machine
        $VM_S$ ← Active $VM_{Hi}$ state as a source
        Migration ($VM_S$, $VM_T$);
    end if
end if
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.
```

We write pseudo code of migration that contain source VM and target VM , consider source minimum load and target subminimum load and apply migration using IMachine and IConsole interface that is API of VirtualBox.

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.
```

EESS implement on base of three criteria i) zero workload i.e. (j<=0) zero workload seen in line no. 2 then exit or return do not start any virtual machine to save energy ii) Minimum workload (j<=J) then PowerON virtual machines and assign VM request to VM use FCFS [16] shown in line number 5,6,7,8,9 and iii)maximum workload (j> =J) line no 12 ,then apply migration of source and target, find VM_s,VM_T in line no 15, 16 explain that find target such as minimum workload or zero workload, if that VM_T is Running state then Pause that VM_T so maximum energy save then apply migration shown in line no 14 .Cloud consumer provide VM request in lease to LMS such as PaaS then apply cloning of virtual machine shown in line 8,9. Update VM_{Hi} set of Host & Guest, j,J, L show in line no 20. end of algorithm in line no 21.

Next Scenario is in EESS steps of PowerON() function using OpenRemotesesssion() function of IWebsessionManager of VirtualBox3.1. If workload is increased such as unbalanced then we apply migration concept for practically implementation of that migration Teleport(), setTeleporterEnabled(), setTeleporterPort(), setTeleporterAddress(), saveSettings() function provide by IMachine, IConsole interface, clone of VM using findHardDisk(), createHardDisk, cloneTo() function of IMedium, IProgress interface of VirtualBox3.1 API.

5. EXPERIMENTAL RESULTS

We present series of experiments to exemplify the outcome of EESS discussed in previous section. The testbed 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) , total time (T), power (W) , number of VM N, number of leases L, VM request from consumer J. We use VirtualBox3.1 [13] API start, stop, pause, resume, clone, and migrate virtual machines in JDK1.7.0 for algorithmic implementation of EESS. We take our EESS, Bully[11] , Round-robin[18], hybrid energy efficient scheduling algorithm[3] results and comparison dissipated in fallowing tables, graphs and explanation.

5.1 Comparison

Number of virtual machines (N): Bully, Round-robin is worst because they wait for job completion and assign new job to VM after completion of, as save energy number of VM are increased then consume energy also increase. Our approach EESS and Hybrid energy efficient scheduling algorithm is always better for energy saving, required less N as shown in Fig.5

Energy Conserved(E): In Fig.6 we are cleared observed that EESS is more energy conserved as compare to hybrid energy efficient scheduling algorithm, but if we elaborate our cloud environment then hybrid energy efficient scheduling algorithm is always better than our approach as per response time and workload utilization, we conserved more energy because of we apply pause, resume of virtual machines, do not directly PowerOFF and PowerON, VM which is happen in hybrid energy efficient scheduling algorithm. Bully approach is better than R-R because of it conserved some amount of energy but R-R is worst about conserved energy which clearly demonstrates in Table 3 as our observation.

Workload Utilization (W) : As compare to workload utilization EESS and hybrid energy efficient scheduling algorithm is better , but we compare our approach to hybrid energy efficient scheduling algorithm observed that hybrid energy efficient scheduling algorithm is comparatively better than EESS which is as shown in Fig.4, Bully is better than round-robin and EESS achieve higher degree of workload distribution.

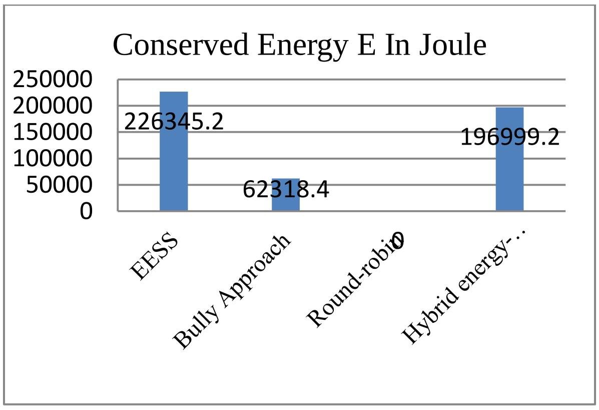

Fig.4 Total conserved Energy E in Joule

Table 2. Detailed Data

Parameters / Methods	Total Energy in J	Total Time in S	Power in W	Number of VM required as per Method N	VM request from consumer	Leases(L)	Energy per Minute in J
EESS	244818	4020	60.9	10	15	8	3654
Bully Approach	408844.8	4368	93.6	14	15	8	5616
Round-robin	471163.2	4944	95.3	15	15	8	5718
Hybrid energy-efficient scheduling algorithm	274164	4092	67	11	15	8	4020

Table 3. Conserved Energy (E) in Joule

Energy (E)In Joule / Methods	Conserved Energy per minute	Conserved Energy per Total time
EESS	2064	226345.2
Bully Approach	102	62318.4
Round-robin	0	0
Hybrid energy-efficient scheduling algorithm	1698	196999.2

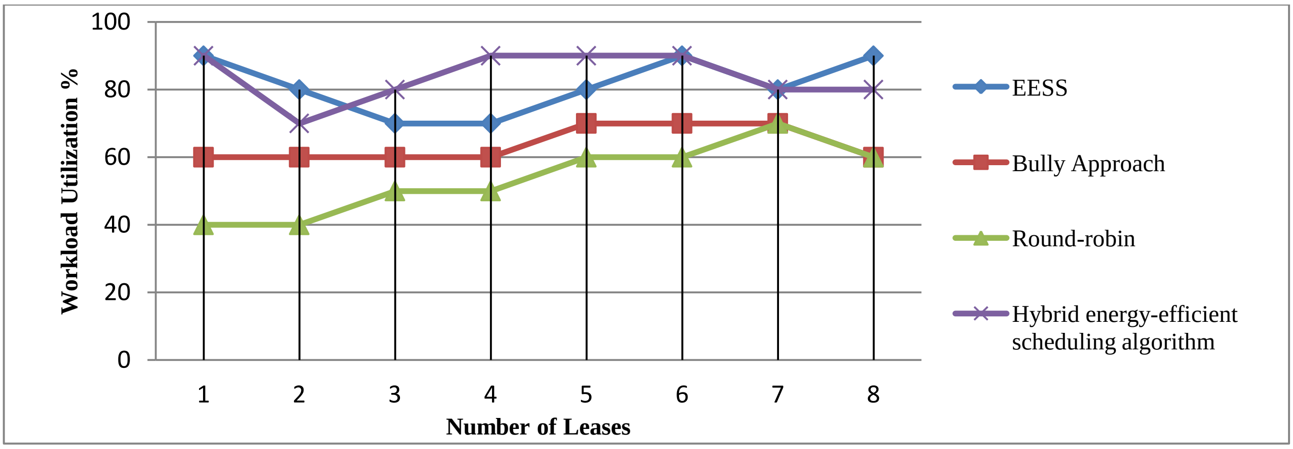

Fig.4 % wise workload utilization as per number of Leases

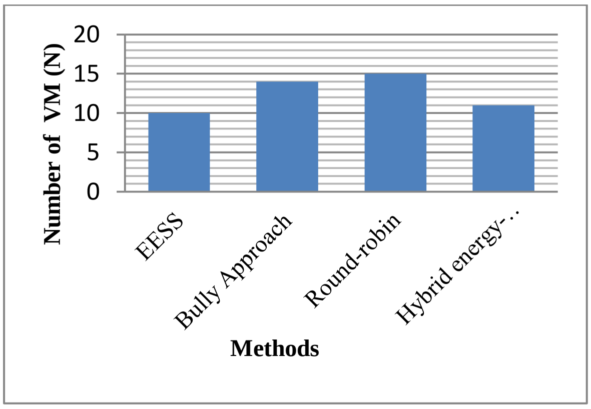

Fig.5 Number of virtual machines required as per method

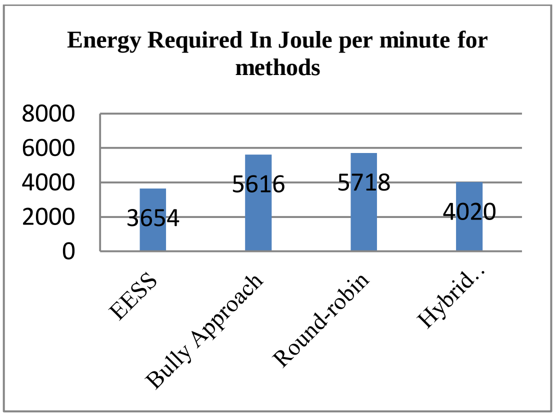

Fig.6 Energy required in joule as per method

6. CONCLUSION

Cloud computing is a new computing paradigm that offers a large quantity of compute and storage resources to the masses. Scientist and startup companies can have access to these possessions by paying a minute amount of money just for what is really wanted. In their various contour and flavors, cloud aim at offering compute, storage network, software, it is combination "as a service". IaaS,PaaS,SaaS are three most common nomenclatures for level of generalization of cloud services, ranging from "raw" virtual servers to sophisticated hosted applications. Virtualization enables high, reliable, and agile deployment mechanisms and management of services, providing on demand cloning, live migration services which improve reliability. A great popularity and apparent success have been seen in this area.

How to provide an energy-efficient scheduling method for the cloud platform has become a challenging problem. Our work is an essential support such application which is publically running, we propose Energy-efficient scheduling scheme call EESS is always beneficial for power generation plant and their survive problems that needs today, and we move towards Green Computing which really needs for us.

7. REFERENCES

[1] Susane Albers (2010),"Energy efficient algorithms", Communication of ACM, vol.53 No.5, 86-96.

[2] Jiandun Li, Junjie Peng, Wu Zhang (2011),"An Energy-efficient Scheduling Approach Based on Private Clouds" , Journal of Information & Computational Science ,volume 8, Number 4 , 716-724.

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

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

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

[6] Chuling Weng, Zhigang Wang, Minglu Li, and Xinda Lu (2009), "The Hybrid Scheduling Framework for Virtual Machine Systems ,"Proc. Conf. VEE09, 113-120.

[7] Gaurav Dhiman ,Giacomo Marchetti ,Tajana Rosing (2009),"vGreen: A System for Energy Efficient Computing in Virtualized Environments ", In conference of ISLPED 2009 San Francisco,California ,USA,19-21.

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

[9] Bo Li, Jianxin Li, Jinpeng Huai, Tianyu Wo, Qin Li,Liang Zhong (2009) , "EnaCloud:An Enegy-saving Application Live Placement Approach for Cloud Computing Enviorments" ,In IEEE Intenational Conference on cloud Computing 200, 17-24.

[10] Aman kansal, Feng Zhao, jie Liu, Nupur Kothari, Arka A. Bhattacharya (2010), "Virtual machine power metering and provisioning", copyright 2010 ACM.

[11] Tami Tamir(2011),"Scheduling with bully selfish jobs"

[12] Pradeep Kumar Sharma, Chandana Das (2012),"Energy efficient scheduling in cloud computing", Inventi Impact: Cloud Computing, Vol. 2012 , Article ID-Inventi:ecc/30/12.

[13] VirtualBox website: www.Virtualbox.org.

[14] Joulemeter website available on :http://research.microsoft.com/en-us/default.aspx

[15] Physics conversation website: www.physics.org.

[16] Scheduling algorithms available on website: http://en.wikipedia.org/wiki/First-come,_first-served.

[17] KVM Project, Kernal based virtual machine available on: http://www.linux-kvm.org

[18] Round robin algorithm available on website: http://en.wikipedia.org/wiki/Round-robin_scheduling

[19] Shailesh Deore, A.N. Patil, Ruchira bhargava (2012),"systematic review of Energy-Efficient scheduling Techniques in cloud computing ", International journal of Computer applications, volume 52, number 15 .

[20] Rajkumar Buyya, James Broberg, Andrzej Goscinski (2011), "CLOUD COMPUTING:Principles and Paradigms " , A Jhon Wiley & Sons, Inc. Publication.

[21] I.Foster (2003), "The grid: Computing without bounds", scientific American , vol.288, No.4,PP.78-85.

[22] R.Buyya, C.S. Yeo, S.Venugopal, J.Brogerg (2009),"Cloud computing and emerging IT platforms: Vision, hype, and reability for delivering comouting as 5th utility", Future generation computer system,25:599-616.

[23] L.M. Vaquero, L.Rodero-Merino, J. Caceres, and M.Lindner (2009),"A break in clouds: Towards a cloud definition", SIGCOMM computer communications Review, 39:50-55.

[24] McKinsey and Co.(2009), "Clearing the Air on Cloud computing " , Technical report.

[25] VMware Inc. VMware high availability(HA), http://www.vmware.com

[26] V. Buge, Y.Kemp, M.Kunze, and G.Quest (2006), "Application of Virtulisation Techniques at a University Grid center", e-Science, 155-155.

[27] 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

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

[29] R.P. Goldberg (1974),"A Survey of virtual machine research",IEEE Computer, 7(6):34-45.

[30] L.Youseff, M.Butrico, D,Da Silva (2008),"Towared a unified ontology of cloud computing", In proceeding of green computing workshop,pp 1-10.