SUCCESSFUL ANALYSIS OF POWER MANAGEMENT SCHEMA IN SECURE CLOUD COMPUTATION PROCESS

E.lakshmi devi,

lakshmideviambiga@gmail.com,

Department of Computer Science and Engineering

Mailam Engineering college, mailam.

A.khomalam

khomalam.mec@gmail.com

Department of Computer Science and Engineering

Mailam Engineering college, mailam.

Abstract - The best natural test today is a worldwide temperature alteration, which is brought about via carbon outflows. Vitality emergency brings green figuring and green registering needs calculations and instruments to be overhauled for vitality proficiency. Green IT alludes to the study and routine of utilizing figuring assets as a part of a productive, compelling and financial way. The different methodologies of the green IT are Virtualization, Power Management, Material Recycling and Telecommuting. The fundamental standards of distributed computing is to make the processing be allocated in an extraordinary number of circulated PCs as opposed to neighborhood PC or remote server. Its closer view is to give secure, snappy, helpful information stockpiling and net processing administration focused by web.

Right now, countless processing frameworks squander an enormous measure of vitality and transmit a lot of carbon dioxide. In this manner, it is important to fundamentally decrease contamination and considerably bring down vitality use. The examination of vitality utilization in distributed computing consider both open and private mists. Distributed computing with green calculation can empower more vitality productive utilization of processing force.

Index Terms- *Cloud Computing, Bee colony algorithm, Ant colony allgorithm, Private and Public Clouds.*

I. INTRODUCTION

This Vision of figuring utilities in view of a serviceprovisioning model foreseen the monstrous change ofthe whole registering industry in the 21st century wherebycomputing administrations will be promptly accessible on interest, likeother utility administrations accessible in today's general public. Similarly,users (buyers) need to pay suppliers just when they get to the processing administrations. Likewise, buyers no more need to put vigorously or experience challenges in building andmaintaining complex IT framework. In such a model, usersaccess administrations in view of their necessities without respect towhere the administrations are facilitated. This model has been alluded toas utility processing, or as of late as Cloud Computing. Thelatter term signifies the framework as a "Cloud" from whichbusinesses and clients can get to applications as administrations fromanywhere on the planet on interest. Henceforth, Cloud processing workmanship server farms [1] that typically utilize Virtual Machine (VM) advancements for combination and environment isolationpurposes .

Distributed computing [2] conveys framework, stage, andsoftware (applications) as administrations, which are made availableto purchasers as sUbscription based administrations under the pay-as you-go model. In industry these administrations are alluded to asInfrastructure as a Service (laaS), Platform as a Service(PaaS), and Software as a Service (SaaS) separately.

Green registering [4] is characterized as the study and routine of outlining , assembling, utilizing, and arranging ofcomputers, servers, and related sub frameworks such asmonitors, printers, stockpiling

gadgets, and systems administration andcommunications frameworks productively and adequately withminimal or no effect on the earth. Research continuesinto key ranges, for example, making the utilization of PCs as energyefficient as could reasonably be expected, and planning calculations and frameworks forefficiency related PC advances. There are severalapproaches to green figuring, to be specific

- Product longetivity
- Algorithmic efficiency
- Resource allocation
- Virtualization
- Power management etc.

Need of green figuring in mists

Modem server farms [3], working under the Cloud computingmodel are facilitating an assortment of utilizations running from thosethat keep running for a few moments (e.g. serving solicitations of webapplications, for example, e-business and interpersonal organizations portalswith transient workloads) to those that keep running for more timeframes (e.g. reproductions or huge information set preparing) onshared equipment stages. The need to deal with different applications in a server farm makes the test of on- ondemandresource provisioning and allotment accordingly totime-changing workloads.

Green Cloud figuring is imagined to accomplish not onlyefficient preparing and usage of computinginfrastructure, but rather likewise minimize vitality utilization. This isessential for guaranteeing that the future development of Cloudcomputing is supportable. Something else, Cloud figuring withincreasingly pervasive front-end customer gadgets connecting withback-end server farms will bring about a colossal acceleration ofenergy utilization. To address this issue, server farm resources[8] should be overseen in a vitality effective way [lO] todrive Green Cloud processing. Specifically, Cloud resourcesneed to be distributed not just to fulfill QoS requirementsspecified by clients by means of Service Level Agreements (SLA), butalso to diminish vitality utilizationArchitecture of a green cloud computing platform

Fig.l, shows the high-level architecture for supporting energyefficientservice allocation in Green Cloud computinginfrastructure. There are basically four main titles involved:

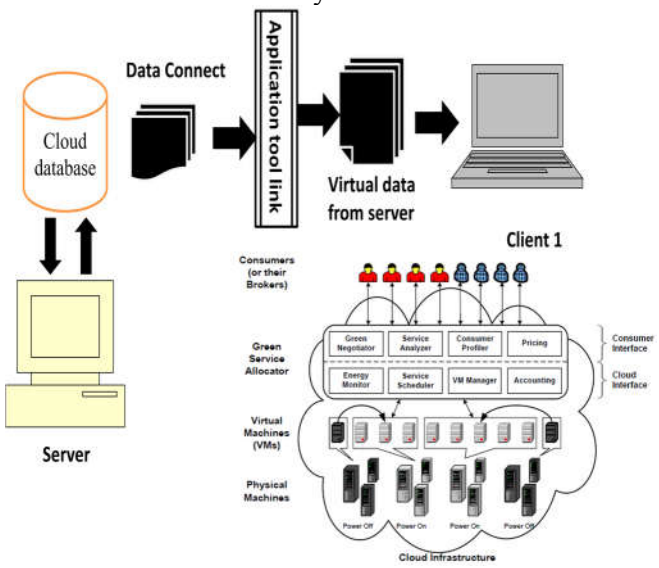

Fig.l: Architecture of a green cloud computing environment

a) Consumers/Brokers: Cloud customers or their brokerssubmit administration demands from anyplace on the planet to theCloud. It is essential to notice that there can be a differencebetween Cloud buyers and clients of sent administrations. Forinstance, a customer can be an organization

sending a Web application, which shows differing workload as per the quantity of clients getting to it. Organizations running applications in the cloud need to comprehend what clients are doing, what the end client experience is similar to and how the fundamental base is performing, Splunk empowers organizations to increase operational insight, investigate issues progressively and oversee applications in the cloud.

b) Green Resource Allocator: Asset designation is utilized to relegate the accessible assets in a financial way. It is a piece of asset administration. In undertaking administration, asset designation is the booking of exercises and the assets required by those exercises while thinking about both the asset accessibility and the task time Acts as the interface between the Cloud framework and customers. It requires theinteraction of the accompanying segments to bolster energyefficient asset administration:

1.GreenNegotiator: Negotiates with the consumers/brokersto finalize the SLA with specified prices and penalties (forviolations of SLA) between the Cloud provider and consumerdepending on the consumer's QoS requirements and energysaving schemes.

2.Service Analyzer: Interprets and analyses the service
requirements of a submitted request before deciding whetherto accept or reject it. Hence, it needs the latest load and energyinformation from VM Manager and Energy Monitorrespectively.

3. Consumer Profiler: Gathers specific characteristics ofconsumers so that important consumers can be grantedspecial privileges and prioritized over other consumers.

4. Pricing: Decides how service requests are charged tomanage the supply and demand of computing resources andfacilitate in prioritizing service allocations effectively.

5. Energy Monitor: Observes and determines which physical machines to power on/off, Monitor system usage, uptimes and other performance metrics. Quickly pinpoint and diagnose issues with flexible search capabilities, Monitor key operational metrics such as end-to-end response times, message queue lengths and failed transaction counts to ensure service levels are met

6. Service Scheduler: Assigns requests to VMs and determinesresource entitlements for allocated VMs. It also decides whenVMs are to be added or removed to meet demand.

7. VM Manager: Keeps track of the availability of VMs andtheir resource entitlements. It is also in charge of migratingVMs across physical machines.

8. Accounting: Maintains the actual usage of resources by requests to compute usage costs. Historical usage informationcan also be used to improve service allocation decisions.

c) VMs: Multiple VMs can be dynamically started and stoppedon a single physical machine to meet accepted requests, henceproviding maximum flexibility to configure various partitionsof resources on the same physical machine to different specificrequirements of service requests. Multiple VMs can alsoconcurrently run applications based on different operatingsystem environments on a single physical machine. In addition,by dynamically migrating VMs across physical machines,workloads can be consolidated and unused resources can beput on a low-power state, turned off or configured to operate atlow-performance levels (e.g., using DVFS) in order to saveenergy.

d) Physical Machines: The underlying physical computingservers provide hardware infrastructure for creating virtualizedresources to meet service demands.

Making cloud computing more green

Mainly three approaches have been tried out to make cloud computing environments more environmental friendly. These approaches have been tried out in the data centers under experimental conditions. The practical applications of these methods are still under study. The methods are:

Resource Allocation or Virtual Machine Migration Techniques:
In a cloud computing environment, every physical machine hosts a number of virtual machines [5] upon which the applications are run. These virtual machines can be transferred across the hosts according to the varying needs and available resources. The VM migration method focuses on transferring VMs in such a way that the power increase is least. The most Power efficient nodes are selected and the VMs are transferred across to them.

Algorithmic approaches: - It has been experimentally
determined that an ideal server consumes[9] about 70% of thepower utilized by a fully utilized server. (See figure 2).

II. PROPOSED FRAME-WORK

So as to add to an eco-accommodating distributed computing, we propose a superior vitality administration for IaaS by center middleware. We recommend that the center middleware ought to have "Power Consumption Policy Maker(PCPM)" "Virtual Machine Turn-on-of Decider(VMTD)", "Administration Assigner to Virtual Machines(SAVM),"Mapping of administration from Virtual Machine to IaaS(MVMI)", "Asset utilizing Recorder(RUR)", "Tracker of Virtual Machines (TVM)","Virtual Machines(VMs)","Service Execution Manager(SEM)" and so forth.. We suggest that the client –level middleware ought to comprises of "Graphical User Interface for Consumer (GUIC)", "Nature of Service Checker(QoSC)", "Administration Accepter(SA)", "Charging System(BS)" and so on. We recommend that the working of general procedure ought to be as per the followin
Step1: At first, consumer would submit services to GUIC, from where service is submitted to SA. SA decides whether to accept the service or not after getting information of Virtual Machine status from TVM. TVM keep tracks about the status of VMs (i.e. whether VMs are available or not to execute the new service) as well as from the information of power consumption policy from PCPM.

Step 2: If service is accepted then it is assigned to QoSC, which fixes up the SLA and the price of the requested service after getting information from BS. BS estimates the cost of a service using the historical data at RUR or by using empirical cost estimation technique. It also decides the penalties for violation of
SLA.

Step 3: After QoSC, services are submitted to SEM, which interacts with PCPM. PCPM interacts with VMTD and turns the required number of VMs on to carry out services and then assign services to VMs using SAVM. RUR keeps resource using record.

Step 4: Finally the active VMs(which are turned on), execute the services on CPUs of IaaS. The active VMs turns the powers of the corresponding CPUs of IaaS on and the rest are made off and then send the services from active VMs to active CPUs of IaaS using MVMI. Thus all CPUs of IaaS are not turned on, only those required are turned on and the rest are turned off. The overall process is shown by the sequence diagram in Figure 1, (Figure 1 is shown at last page). In this way power consumption of energy is minimized along with minimization of operational cost and the emission of carbon dioxide gas to the environment. Thus cloud computing is a very promising technology due to its eco-friendly nature with minimized operational cost.

III. PROPOSED ALGORITHM

In nature, we discover a considerable measure of intriguing aggregate conduct of honey bees, ants, and so forth. It is watched that bumble bees [17] work in a decentralized and self –organized way. The

nectar settlements hone division of work between the falsifiers, who work in the field gathering the nectar and the sustenance stores who work in the hive to store the nectar. The counterfeiter honey bees look the nectar aimlessly and in the wake of recovering the nectar they come back to the hive and begin moving on the move floor keeping in mind the end goal to spur the devotee falsifiers to get to the nectar. The term of this move is firmly associated with the inquiry time experienced by the moving honey bees. In this paper, we have utilized this conduct of honey bees to articulate distributed computing minutely. In distributed computing environment, we watch that some CPUs of IaaS are over-burden for preparing shoppers' administrations, some are under stacked and some are thoroughly sit without moving. We can spare the utilization of vitality by killing these unmoving CPUs and rescheduling administrations from over-burden CPUs to under stacked CPUs. Our proposed strategy won't just spare vitality yet will likewise counteract potential SLA infringement. Be that as it may, the issue lies in dealing with these unmoving CPUs, over-burden CPUs and under stacked CPUs adequately; for this we have proposed Bee-Ants settlement framework. Toward the starting, we separate our employments into two sections; the principal part, which takes care of the correct administration of over-burden and under stacked CPUs (administration rescheduling) and the second part, which deals with the unmoving CPUs (power utilization administration). We propose honey bee settlement calculation for administration rescheduling and subterranean insect state calculation for force utilization administration. Therefore for administration rescheduling, we consider TVM as hive which comprise of honey bee specialists, from where operators begin to manufacture for nectar. Here nectar suggests the limit estimation of CPU. Low quality of nectar infers lower limit estimation of CPU though great nature of nectar infers the higher edge estimation of CPU. In this manner great nature of nectar suggests over-burden CPU and low quality of nectar infers under stacked CPU, what's more the move floor speaks to the administration booking table, where the data about the status of the CPUs that the honey bee operators have gone by, are put away . In this way the fashioning honey bees in the wake of finding the nectar, impart specifically through moving on the move floor. There are two sorts of move; waggle move which suggests low quality of nectar and tremble move (round move) which infers great nature of nectar. Subsequent to moving, the old honey bee operators bite the dust. On the off chance that the move is tremble move, then the new conceived honey bee specialists fly to gather the great quality nectar and store them in the hive. After this operation the old honey bee specialists bite the dust and the new conceived honey bee operators begin to fly with the great quality nectar put away in the hive, lastly blend them with those sources which are holding low quality nectar. This procedure of appropriation goes ahead until there is a uniform nature of nectar in every one of the sources. Really this procedure is actualized for rescheduling the administrations from over-burden CPUs to under stacked CPUs keeping in mind the end goal to give great QoS(Quality of Services) to purchasers. Again if the moves of the honey bee operators are waggle move then hive TVM states SA to acknowledge new administrations from customer. After this stride, we apply subterranean insect province based calculation to locate the unmoving CPUs and after that turning them off so as to minimize the utilization of force and henceforth bring down the operational expense. The subterranean insect settlement from VMTD begin to discover the wellspring of sustenance, in our connection it is unmoving CPUs. The subterranean insect state calculations were presented by Dorigo(1992)[18] as a disseminated meta-heuristic, which imitates the social manufacturing conduct of genuine ants. The calculation includes simulated operators called ants who demonstrate a helpful conduct to locate the most limited way to the sustenance source from their home.

Subterranean insect System [18] is most appropriate in tackling issues which keep running in dynamic and fluctuated situations without the assistance of any focal control. This is the reason it is all that much material to disseminated critical thinking. As distributed computing is alterable in nature, consequently we propose to apply subterranean insect framework in distributed computing in discovering unmoving CPUs. Because of their confined perceivability, ants connect with each other by dropping a concoction known as pheromone for their backhanded and worldwide correspondence. The pheromone focus is conveyed on the way navigated by the ants and is slanted to move over the way with higher pheromone fixation. Along these lines, ants based calculation develop an answer in an iterative way by getting data from their short perceivability (nearsighted data) and pheromone fixation conveyed on way (long haul data). A dissipation instrument is acquainted all together with

maintain a strategic distance from stagnation. Based upon the above dialog, the calculations based eco-accommodating distributed computing are given underneat

```
Procedure GUIC (service)

Begin

        Boolean flag

        flag← SA(services)

        if (flag = 0)

        then

        write("Request Service can not be carried out")

        else

        write("Request Service is accepted")

End


Function Boolean SA (service)

Begin

        Boolean flag.

        flag← TVM(Request for checking of availability of Virtual Machines)

        If (flag =0)

        Return(false)

        else

        do

        call Procedure VMTD()

        Return (true)

end

End

Function Boolean TVM ( Request_for_Checking_of_

_Resource_Avablity)

Begin

        Boolean flag
```

Bee stores in hive.

Forger Bees from hive start to search for nectar

i.e. Availability of VMs.

Forger Bees return back and start dancing on

the dancing floor.

If (dance be Waggle Dance)

then Return (true.)

do

Call QoSC (service)

Return (True)

End

Procedure QoSc (service)

Begin

Ψ is upper utilization of thresholds of CPUs representing good quality of nectar.

δ is lower utilization of thresholds of CPUs representing poor quality of nectar.

Repeat until Ψ = δ do

After observing the Tremble dance on the dancing floor, new born bee agents start to collect good quality nectar from the source and start to store them in the hive. Gradually previous sources of good quality nectar, loose their quality, hence Ψ = Ψ -1

After this operation the old bee agents die and the new born bee agents start to fly with the good quality nectar stored in the hive, and finally mix them with those sources which are holding poor quality nectar. This process of distribution goes on until there is a uniform quality of nectar in all the sources, hence δ = δ + 1.

done.

End.

Procedure VMTD ( )

Begin

/* Initialization*/

For every edge (i, j) do

$u_{ij}(o) = \iota_0$

$\eta_{ij} = \eta_0$

End For.

For k =1 to m do

Place k on nest M

End for

/* Main Loop */

For t = 1 to tmax do

For k = 1 to m do

Build tour $T^k_{(t)}$

Choose the next CPU j with probability

$$p^k_{ij} = \frac{[\iota_{ij}]^\alpha [\eta_{ij}]^\beta r_{ji}(d)}{\Sigma [\iota_{ij}]^\alpha [\eta_{ij}]^\beta}, \text{ if } j \varepsilon N^k_i$$

$l \varepsilon N^k_i$

= 0, otherwise.

Find the threshold ξ_j of the j^{th} CPU

If $|\xi_i < \varepsilon|$ then turn that CPU off.

Where i is the current CPU.

$\eta_{ij} = 1/d_{ij}$, called visibility

d_{ij} = Euclidian distance.

ι_{ij} = Pheromone Trials.

α = Parameter which determines the relative influence of the pheromone trial.

β = Parameter which determines the relative influence of the visibility.

ε is very very small quantity.

N^k_i is the set of all CPUs that the ant still has to visit when it is on CPU i. By exploiting N^k_i an ant k can avoid visiting a CPU more than once.

M is the Nest from where ants start to turn off those CPUs whose thresholds are very small.

End For.

/* Updating Pheromone Trails*/

For every stage (i , j) do
Update pheromone trails by using following:

$$\tau_{ij} = (1 - \rho) \cdot \iota_{ij}(t) + \sum_{k=1}^{m} \Delta_\iota{}^k_{ij}$$

Where $\Delta_\iota{}^k_{ij} = 1 / C^k$, if stage (i, j) ε T^k
$= 0$, otherwise

c^k is the sum of lengths of arc belonging to T^k
ρ is the pheromone evaporation rate.
End.

IV. EXPERIMENTAL RESULT

A private cloud has been setup based on Ubuntu's 10.04 Server edition that consists of two Servers – Server A and Server B. Server A acts as the cloud, cluster, warehouse and storage controller and Server B acts as node controller. We configured Machine A on a Core2duoX6800 processor based machine with 2GB DDR 2 RAM and 80 GB Hard disk. Machine B is running on an AMD PhenomeII X4 965 processor with 4 GB DDR 3 RAM and 250 GB Hard disk. The nodes communicate through a fast local area network. This paper demonstrates the results of local accessing of multiple web services running on the web servers. Experiments have been carried out on above mentioned cloud environment. In order to execute the web services on Machine A, we have used the following commands:

```
If [ ! –e ~/.euca/mkey.priv]; then

mkdir –p –m 700 ~/.euca

touch ~/.euca/mykey.priv

chmod 0600 ~/.euca/mykey.priv

euca-add-keypair mykey > ~/.euca > ~/.euca/mykey

fi
```

Thus creating key pair, we now log into our instance as root. After that, we can monitor state of instance using the command:

```
watch –n5 euca-describe-instances
```

Thus we start to access different web services. In order to evaluate our algorithm, we have conducted several experiments with different values of the energy consumption and SLA violation. We observe that our experiments shows that with rise of consumption of energy, there is a sharp fall in SLA violation but the beauty of our algorithm is that it estimates some cases in which there are

optimal values between energy consumption and SLA violation. The fact is clearly shown in Figure 2. In Figure2, from left to right, we are having some lines; indicating minimum energy consumption, mean energy consumption, standard
deviation and maximum energy consumption.

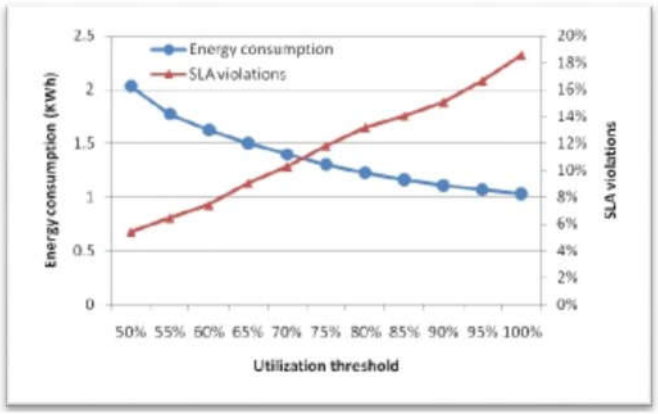

Figure 3.

Thus the algorithm states some cases in which we can have better utilization of energy with less SLA violation.

V. CONCLUSION

In this attempt we have stressed on turning so as to add to an eco-accommodating distributed computing the switches of unmoving CPUs of IaaS off, all the while we have concentrated on keeping away from SLA infringement for minimized vitality utilization and operational expense. For the general procedure, we have proposed calculation. We feel that a legitimate security administration is required for eco-accommodating distributed computing, which we expect to explain in our prospective try. The costfunctions joined into these heuristics successfully catch vitality sparing potential outcomes and their capacity has beenverified by my assessment study. The outcomes in this studyshould not have just an immediate effect on the decrease ofelectricity bills of cloud base suppliers, however alsoimply conceivable savings(with better asset provisioning) inother operational expenses (e.g., rent for floor space). Of course,the lessening in the carbon foot shaped impression of mists is anotherimportant spinoff.

REFERENCES

[1] L.Vaquero, L.Rodero-Marino,J.Caceres, M.Linder, “A break in the clouds: towards a cloud definition”, SIGCOMM Computer Communication Review,2009, pages 137-150

[2] T.D. Seely, “The Wisdom of the Hive”, Harvard University Press, 1995, pages 295-296.

[3] Marco Dorigo and Thomas Stutzle “Ant Colony Optimization”, Prentice-Hall of India, India, 2001

[4] U.S. Environmental Protection Agency. Report to congress on server and data center energy efficiency. Technical report, ENERGY STAR Program, Aug. 2007.

[5] A. Berl, E. Gelenbe, M. di Girolamo, G. Giuliani, H. de Meer, M.-Q. Dang, and K. Pentikousis. EnergyEfficient Cloud Computing. The Computer Journal,53(7), September20 1 O,doi: 10.10931 comjnl!
bxp08

[6] Ministry of Economy, Trade and Industry,
Establishment of the japan data center council, Press Release.

[7] The green grid consortium (2011).

[8] A.Beloglazov and R. Buyya, Energy Efficient
Resource Management in Virtualized Cloud Data Centres, Proceeding of the 10th IEEEI
ACM International Conference on Cluster,Cloud and Grid Computing, 2010.

[9] R. Basmadjian, N. Ali, F. Niedermeier, H. d. Meer and G. Giuliani, A Methodology to Predict the Power Consumption for Data Centres, Proceedings of eEnergy 2011.

[11] http://en.wikipedia.org/wiki/Clouds

[12] EMC 2008 Annual overview Releasing the power of infonnation, http://www.emc.comJ digital_universe.

[13] R. Bianchini and R.Rajamony, "power and energy management for server systems," IEEE Computer, voI.37,no. ll,pp.68-74,2004.

[14] S.Albers, "Energy -Efficient Algorithms,"
Communications of the ACM, vo1.53, no.5, pp.86- 96,May 2010.