Energy-Aware Manipulated Framework for Power Calculation in Cloud Datacenters to Condense Power Consumption

Rajesh P.Patel [1], Dr. Ramji Makawana[2]
[1] Ph.D Scholar, Computer Engineering, [2] Founder, e-Vsn Technology, Rajkot.
[1]C. U. Shah University, Wadhwan, India.

Abstract: Cloud computing is a type of service computing that relies on shared computing resources which are available on-demand and can be rapidly provisioned and released with minimal management effort. Computing resources of datacenter's like servers, network and cooling system which consistently grows in size, complexity also consume more power. To resolve inefficiency in data center and server frameworks this proposed work takes power consumption into account as a key parameter of the datacenter's configuration. To lower the power consumption while fulfilling performance requirements this work proposed an energy-aware framework which differentiate Best and UnBest datacenter using power calculation in Cloud Datacenters. The proposed energy-aware allocation heuristics provision datacenter resources to client applications, calculate the power consumption, finally identify the Hot and Cold Datacenters and find status of Virtual Machines for efficient migration in data centers.

***Keywords*:** **Server Consolidation, VM Migration, Quality of Service, Virtualized Data Center, Service Level Agreements, Highest Thermostat Setting**

I. INTRODUCTION

Cloud computing is architecture for providing computing service via the internet on demand and pay per user access to a pool of shared resources namely networks, storage, servers, services and applications, without physically acquiring them [1]. This type of computing provides many benefits for businesses, shorter start-up time for new services, lower maintenance and operation costs, higher utilization through virtualization, and easier disaster recovery that make cloud computing an attractive option [2]. This technological trend has enabled the realization of a new computing model, in which resources (e.g., CPU and storage) are provided as general utilities that can be leased and released by users through the Internet on-demand fashion [3]. Multi-Tenancy in Cloud Computing occurs when multiple consumers share the same application, running on the same operating system, on the same hardware, with the same data-storage system and both the attacker and the sufferer are sharing the common server [4]. Cloud computing is associated with service provisioning, in which service providers offer computer-based Services to consumers over the network [5].

Cloud computing is one of the internet based service provider which allows users to access services on demand. It provides pool of shared resources of information, software, databases and other devices according to the client request [6]. Cloud computing provides various services according to the client request related to software, platform, infrastructure, data, identity and policy management [7]. Delivering model in cloud environment states in three main types; Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [8].

In cloud environment the services requested by the client is rectified by employing virtual machines present in a server. Each virtual machine has different capabilities, so it becomes more complex to schedule job and balance the work-load among nodes [9]. Load balancing is one of the central issues in cloud computing it is a mechanism that distributes the dynamic local workload evenly across the entire server in the whole cloud to avoid a situation where some servers are heavily loaded while others are idle or doing little work [10].The trend towards server-side computing and the exploding popularity of Internet services has made data centers become an integral part of the Internet fabric rapidly. Data centers become increasingly popular in large enterprises, banks, telecom, portal sites, etc, [11]. As data centers are inevitably growing more complex and larger, it brings many challenges to the deployment, resource management and service dependability, etc. [12].

The rest of the paper is organized as follows. Section-II contains the survey of related work. Section-III describes proposed framework by culling energy aware data centers. Section-IV, explains various dynamic virtual machine consolidation algorithms proposed by researchers. Section-IV, describes results and discussion. Section-V concludes the work.

II. RELATED WORK

Nazneen Taj et.al [13] proposed a work on energy calculation in a datacenter with the Losses and the Method calculating the energy losses that occur during scheduling also the losses caused by the components used. They have proposed equations to calculate the total energy consumed. They have discussed how energy supplied at different times or different intervals of time can be saved where some measures have been proposed to minimize the losses.

Soumya Ranjan Jena et.al [14] proposed a four different power models such as linear model, cubic model, square model and square root model on an Infrastructure-as-a-Service (IaaS) cloud environment to find out the best one. They found overall CPU utilization, overall power consumption in each case by migration, relationship between CPU utilization and overall power consumption in each model. It has found best model out of these four in a predetermined time period based on total power consumption and at the end calculate the accuracy through R-squared and mean square error. They found that the cubic polynomial model is the most efficient one and consume less power.

Giuseppe Portaluri et.al [15] presented a set of power-aware dynamic allocators using software defined networking paradigm for virtual machines in cloud data centers (DCs). Each VM request is illustrated by four parameters: 1) CPU; 2) RAM; 3) disk; and 4) bandwidth. Many VM request are accepted by the allocators designed considering the power consumption of the network devices. The allocators are differentiate based on their allocation policy (best fit/worst fit), allocation strategy (single/multi objective optimization), and joint/disjoint selection of IT and network resources. They have evaluated the behavior of all possible pairs policy-strategy by changing the load of the DC and the number of VMs to be allocated. Authors have introduced ten different allocation policies, and compare them with a baseline which is made of first available server (first fit). The experimental result shows that joint approaches outperform disjoint ones.

Ayan Banerjee et.al [16] proposed a coordinated cooling-aware job placement and cooling management algorithm, in which HTS was aware of dynamic behavior of the Computer Room Air Conditioner (CRAC) units and places jobs to reduce cooling demands from the CRACs. HTS also dynamically updates the CRAC thermostat set point to reduce cooling energy consumption. Further, the Energy Inefficiency Ratio of SPatial job scheduling algorithms, also referred as SP-EIR, was analyzed by comparing the total (computing + cooling) energy consumption incurred by the algorithms with the minimum possible energy consumption, while assuming that the job start times were already decided to meet the Service Level Agreements (SLAs). This analysis was performed for two cooling models, constant and dynamic, to show how the constant cooling model assumption in previous research misses out on opportunities to save energy. Simulation results based on power measurements and job traces from the ASU HPC data center show that: (i) HTS has 15% lower SP-EIR compared to LRH, a thermal-aware spatial scheduling algorithm; and (ii) in conjunction with FCFS-Backfill, HTS increases the throughput per unit energy by 6.89% and 5.56%, respectively, over LRH and MTDP (an energy-efficient spatial scheduling algorithm with server consolidation).

Demetrio Lagana et.al [17] power supply dominates the operational costs of cloud infrastructure and many solutions are proposed to reduce these operational cost. In this paper, authors proposed a complex infrastructure which made up of data centers (DCs) placed in different geographical areas where renewable energy generators are co-located with data center and they are installed to reduce the amount of energy that must be purchased by the power grid. Authors have considered EcoMultiCloud, which is very flexible and is suited to the considered scenario. A load management strategy for multi-objective load management strategies; they have tailored it to the presence of renewable energy sources. By considering both energy cost variations and the presence of renewable energy production, cost reduction is achieved in the load allocation process, when virtual machines (VMs) are assigned to a data center of infrastructure. Performance is analyzed for specific infrastructure having four data centers. Despite being intermittent and highly variable, renewable energy can be effectively broken in geographical data centers when a smart load allocation strategy is implemented.

Gaurav Chadha et.al [18] illustrated LIMO, a runtime system that dynamically manages the number of running threads of an application for maximizing performance and energy-efficiency. LIMO observes threads progress also the usage of shared hardware resources which help to determine the best number of threads to run and the voltage and frequency level. LIMO gives an average of 21% performance improvement and a 2x improvement in energy-efficiency on a 32-core system over the default configuration of 32 threads for a set of concurrent applications from the PARSEC suite, the Apache web server, and the Sphinx speech recognition system.

Jordi Guitart et.al [19] proposed an overload control strategy for secure web applications that brings together dynamic provisioning of platform resources and admission control based on secure socket layer (SSL) connection differentiation. Dynamic provisioning enables additional resources to be allocated to an application on demand to handle workload increases, while the admission control mechanism avoids the server's performance degradation by dynamically limiting the number of new SSL connections accepted and preferentially serving resumed SSL connections (to maximize performance on session-based environments) while additional resources are being provisioned. It demonstrates the benefit of the theme of this work for efficiently managing the resources and preventing server overload on a 4-way multiprocessor Linux hosting platform, especially when the hosting platform was fully overloaded.

Anton Beloglazov et.al [20] proposed an architectural framework and principles for energy-efficient Cloud computing. Based on this architecture, open research challenges, resource provisioning and allocation algorithms for energy-efficient management of Cloud computing environments. The proposed energy-aware allocation heuristics provision data center resources to client applications in a way that improves energy efficiency of the data center, while delivering the negotiated Quality of Service (QoS). In particular, this paper conduct a survey of research in energy-efficient computing and propose: (a) architectural principles for energy-efficient management of Clouds; (b) energy-efficient resource allocation policies and scheduling algorithms considering QoS expectations and power usage characteristics of the devices; and (c) a number of open research challenges, addressing which could bring substantial benefits to both resource providers and consumers. This work was validated the proposed approach by conducting a performance evaluation study using the CloudSim toolkit. The results demonstrate that Cloud computing model had immense potential as it offers significant cost savings and demonstrates high potential for the improvement of energy efficiency under dynamic workload scenarios.

Nadjia Kara et.al [21] proposed to address the issues for a specific application of IVR. It defines task scheduling and computational resource sharing strategies based on genetic algorithms, in which different objectives are optimized. The purpose of chose genetic algorithms because of its robustness and efficiency for the design of efficient schedulers has been largely proven in the literature. More specifically, this method identifies task assignments that guarantee maximum utilization of resources while minimizing the execution time of tasks. This paper also proposes a resource allocation strategy that minimizes substrate resource utilization and the resource allocation time. Also this method simulated the algorithms used by the proposed strategies and measured and analyzed their performance.

Antonio Corradi et.al [22] illustrated the problem of VM consolidation in Cloud scenarios, by clarifying main optimization goals, design guidelines, and challenges. To better support the assumptions in this paper, it introduced and used Open Stack, an open-source platform for Cloud computing that was now widely adopted both in academia and in the industrial worlds. Our experimental results convinced us that VM consolidation was an extremely feasible solution to reduce power consumption but, at the same time, it had to be carefully guided to prevent excessive performance degradation. This paper had shown that performance degradation is not easy to predict, due to many entangled and interrelated aspects. At this point, this work interested in going on investigating other important research directions. First, it wants to better understand how server consolidation affects the performance of single services and the role of SLAs in the decision process. The goal along that direction was the automatic identification of meaningful service profiles useful to detail introduced workload, e.g., either CPU or network bound, to better foresee VM consolidation interferences. Second, it wants to deploy a larger testbed of the OpenStack Cloud, so as to enable and test more complex VM placement algorithms. Third, it wants to extend the management infrastructure to perform automatic VM live migration, in order to dynamically reduce Cloud power consumption.

III. ENERGY AWARE FRAMEWORK IN CLOUD BY CULLING ENERGY AWARE DATACENTER.

The proposed methodology consumes two novel methods, such as Energy aware datacenter selection to improve the performance in cloud and Server consolidation to utilize the power consumption in servers present in cloud. In a cloud various datacenters are requested by clients during task processing. Data centers are powerful ICT (Information and communication technology) facilities that consume more power due to more number of tasks. To lower the power consumption while fulfilling performance requirements we propose a *flexible* and *energy-aware* framework for the (re)allocation of virtual machines in a data center. As a basis, this model use a component called "Power Calculator", which is also developed within the FIT4Green project. When provided with a description of the datacenter physical and dynamic elements, this component is able to simulate the power consumption of every part of the data center on a very fine level of granularity , in real time. The proposed energy-aware allocation heuristics provision data center resources to client applications in a way that improves energy efficiency of the data center, while delivering the negotiated Quality of Service (QoS).

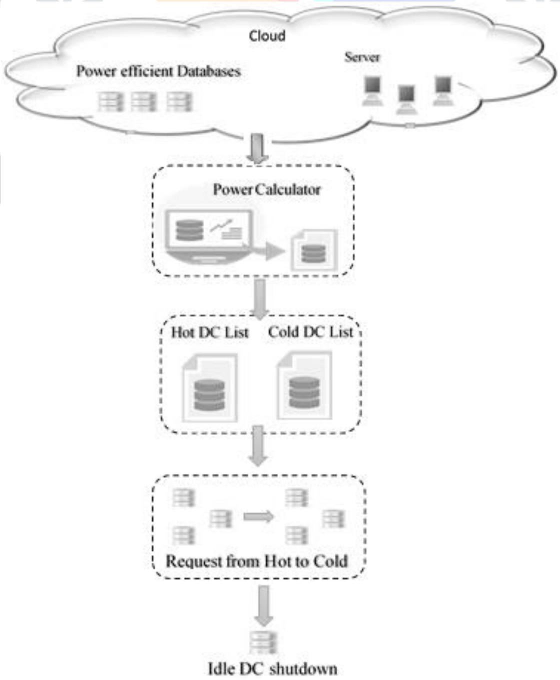

Figure 1: Proposed Frame work

3.1 Culling energy aware DC

In a cloud various datacenters are requested by clients during task processing. Data centers consume more power due to more number of tasks. To lower the power consumption as well as fulfilling performance requirements, a flexible and energy-aware framework 'Energy aware datacenter's is proposed. As a basis, this model use a component called "Power Calculator", which is developed as a part of Green project. This component provided description of the datacenter physical and dynamic elements, this description illustrates power consumption of the data center.

Physical element in a data center involves its memory and loading capacity. Dynamic elements involve the amount of request. From this description the hot and cold datacenters are list out. According to this description the load from hot datacenters moves to the cold. Thus the extra power consumed by the hot data center decreased also the number of request decreased by completion of request thus by this action the number of idle datacenter increased. Based on the monitoring of power calculator the idle data centers are shut down. So the power consumed by the datacenters decreased.

3.1.1 DC Power consumption by its physical components (Network Switches, Cooling System, Racks and Servers)

A datacenter in Fig.1 comprises a structure typically as: Network Switches, Cooling System, Racks and Servers depicts a datacenter configuration with racks. Each rack contains several servers so that, each server has a dedicated power unit and a cooling fan.

The used datacenter has several rack-based cooling systems that are dedicated to each rack. In this structure, the cold air is delivered directly inside the rack and heat generated by the servers is transferred through heat risers to cooling rack. Hence, the efficiency of cooling system increases since the exact location of the air conditioner to the target load is determined.

The used network topology in the most of datacenters is a tree structure; this is shown in Fig. 1. In this case, several top of rack switches connect to one aggregate switch and, in the higher level, the aggregation-layer switches connect to a core router in order to track and route the migrated VMs from one sever to another server in another location.

Total power consumption of datacenter contains consumed computation power by servers (Eq. (1)), cooling system and, network components. To this end, the summation of power of all ON switches in each layer of network topology forms the power consumption of related layer. Therefore, the consumed network power (P_{NET}) is expressed as follows:

$$P_{NET} = P_{ToR} + P_{Agr} + P_{CR} \tag{1}$$

P_{ToR}, P_{Agr} and P_{CR} denote power consumption of top of rack switches, aggregation-layer switches and core router respectively. For total cooling power consumption (P_c) if j^{th} rack is turned on, the related cooling rack will be turned on and consumes P_j power. Therefore, P_c is defined as

$$P_c = \sum_{j=1}^{N_r} P_j \tag{2}$$

Where N_r determines total number of racks. Finally, the total power consumption of datacenter P_{DC} is defined as follows:

$$P_{DC_i} = P_S + P_C + P_{NET} \tag{3}$$

$$P_S = \sum_{i=1}^{N_s} P_i \tag{4}$$

Where P_S represents total power consumption of servers and N_S specifies the total number of servers in all racks. From eq. (3) the power consumption of each DC was evaluated. After finding the power consumption of each DC then the data centers are differentiated based on Maximum and Minimum power consumption.

3.1.1.1 Maximum Power consumption of DC

Maximum power consumption DCs are separated if the Power of that DC is greater than the limit.

$$maxP_{DC_i} = P_{DC_i} > L \tag{5}$$

3.1.1.2 Minimum Power consumption of DC

Minimum power consumption DCs are separated if the Power of that DC is less than the limit.

$$minP_{DC_i} = P_{DC_i} < L \tag{6}$$

The limit (L) employed here for evaluating maximum and minimum power consumption of DC is calculated by Normal power consumption of DC (NP_{DC}) multiplied by power required for a process $(P_{process})$ divided by Number of process $(N_{process})$

$$L = \frac{NP_{DC} \times P_{process}}{N_{process}} \tag{7}$$

3.1.2 Dynamic components of DC

Total Number of Requests (R) is collected from GIS

$$R = GIS \tag{8}$$

The utilized capability of the DC_i is calculated using eq. (5):

$$UC_{DC_i} = \frac{OC}{ET} \quad (9)$$

OC- Original Capacity, ET- Execution Time

For all the jobs in the queue *WT* from *total Request R*. The difference of estimated execution time EET and the completed execution time CET is computed using eq. (6):

$$WT = EET - CET \quad (10)$$

The execution time *ET* of the job determines the speed of executing a job *T*. It can be calculated by finding difference between the current time and entry time of job into VM using eq. (7).

$$ET = Current\,time - Entry\,time\,of\,job\,into\,VM \quad (11)$$

The completion time of the job is sum of waiting time *WT* and execution time *ET*

$$CT = WT + ET \quad (12)$$

3.1.2.1 Estimated completion time

Estimated completion time of an un-started job is calculated by estimating the number of floating-point operations (supplied by the project) and floating-point benchmark of a CPU.

$$ECT = FA + (1 - F)B \quad (13)$$

Where F is the fraction done, A is the estimate based on elapsed CPU time and fraction done, and B is the estimate based on benchmarks and floating-point count

3.1.2.2 Best DC

Best DCs are the DC with less computation time. If the completion time of a allocated process is lesser or equal to the estimated completion time then it is chosen as best DC

$$BDC_i = CT \leq ECT \quad (14)$$

3.1.2.3 UnBest DC

Unbest DCs are the DC with high computation time or crashed DCs. If the completion time of a allocated process is higher than estimated completion time then it is chosen as unbest DC

$$UDC_i = CT > ECT \quad (15)$$

3.1.2.4 Hot DC

It is evaluated by utilizing a comparative performance servers present in the list of both max power consumption and time utilization. If a server presents in both max power and unbest DC then it is concluded as hot DC (H_{DC_i})

$$H_{DC_i} = maxP_{DC_i} \cap UDC_i \quad (16)$$

3.1.2.5 Cold DC

If a server present in both min power and best DC then it is concluded as cold DC (C_{DC_i})

$$C_{DC_i} = minP_{DC_i} \cap BDC_i \quad (17)$$

After listing the hot and cold Datacenters separately further new request from users are moved from hot DC to cold DC.

3.2 Power saving Scheme (Idle shut down)

In a first step we have grouped all servers s_i into families s_k that share similar characteristics, where $i \in I$ is the index of the server in the data centers, and $k \in K$ is the index of the family. The VMs v_i are also grouped into V_l families that share similar characteristics, where $j \in J$ is the index of the VM in the data centers, and $l \in L$ is the index of the family. Note that such an assumption is possible since it is common for a data centre to have families of similar equipment and because VMs often share similar run-time characteristics as well.

Furthermore we have defined a vector H_i = <h_{i1}, ... , h_{ij}, ... , h_{in}> for each server s_i that denotes the set of VMs assigned to that server, where h_{ij} = 1 if the node s_i is hosting the VM v_j and 0 otherwise. The whole array *H* therefore represents how the

VMs are assigned on servers in the different data centers. Now, the power consumed by server *I* depends on its physical components as well as on the set of VMs present:

$$P_i = f(S_i, H_i \bullet v) \tag{18}$$

Here, *f* is the power consumption function provided by the Power Calculator. The dot is the vectorial product and *v* is the vector of all VMs. Thus, $H_i \bullet v$ is the vector representing all the VMs that are located on server *i*.

Next, we extend the function by a factor representing the fact that, if there are no VMs on a server, it can be switched off meaning that it is not consuming any energy any more. For this purpose let

X_i be a variable which has a value of 1 if there is at least one VM in a server *i*, 0 otherwise:

$$X_i = \begin{cases} 1, \exists j \in J | h_{ij} = 1 \\ 0, otherwise \end{cases} \tag{19}$$

Here the Idle Servers of a datacenter are sorted out then they are shut down to save unwanted power consumption of those Datacenters.

IV. RESULTS AND DISCUSSION

This section evaluates the proposed approach on a cloud tested Cloudsim inspired by a corporation. The proposed method evaluates the power consumption of datacenters and Virtual machines in Cloudsim toolkit using Netbeans IDE.

4.1 Performance evaluation of proposed work in Datacenters

The power consumed by the datacenters is evaluated by validating the proposed approach in an environment as close as possible to a cloud data centre. The evaluation of power consumption of datacenter can be executed by considering datacenter components like Servers, Network component. Then minimized power consumption by employing Migration of new request from Hot DC to Cold DC is evaluated.

In Cloudsim we have calculated the power consumption of datacenters using power calculator. User requests are assigned and workloads are assigned to each DC servers. We have bifurcated the Cold and Hot Datacenters based on Best and UnBest datacenters having low computation time and higher computation time also based on minimum power consumption and maximum power consumption respectively. Virtual machines are created and assigned to datacenter host machines. Based on power calculation of virtual machines the virtual machines are decided Busy or Idle status which helps for migration. In our experiments we consider 5 datacenters.

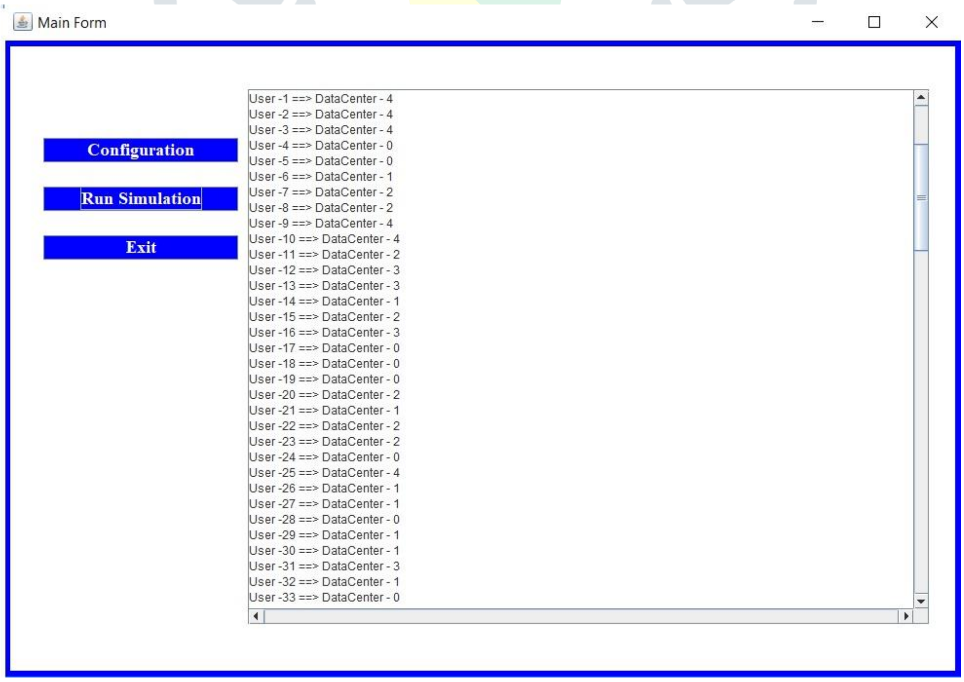

Figure 2: Users assigned to Datacenters (DCs)

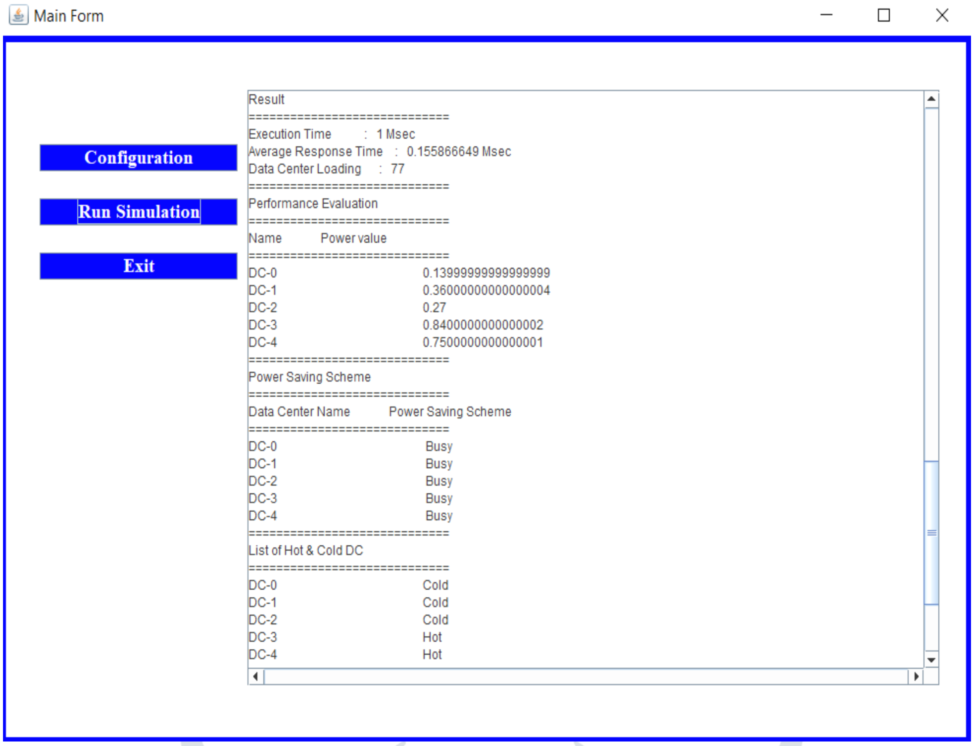

Figure 3: Power consumption & List of Hot & Cold DCs

Table 1: Datacenters Status

Data center	Status (Based on Power consumption)
DC-0	Cold
DC-1	Cold
DC-2	Cold
DC-3	Hot
DC-4	Hot

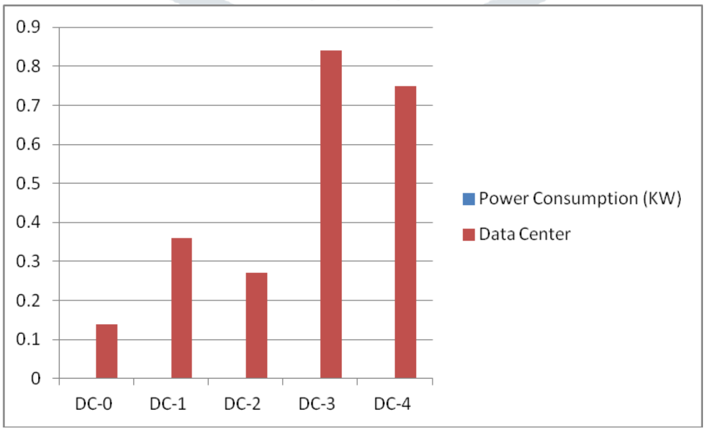

Figure: 3 Power Consumption of Datacenters

Table 2: Virtual Machine Status for Migration

Virtual Machine Status			
Virtual Machine	**Status**	**Power Consume(W)**	**Memory (MB)**
VM-0	Idel	5.024409	502
VM-1	Busy	10.26621	492
VM-2	Busy	10.80642	492
VM-3	Busy	10.32794	492
VM-4	Idel	5.689816	502

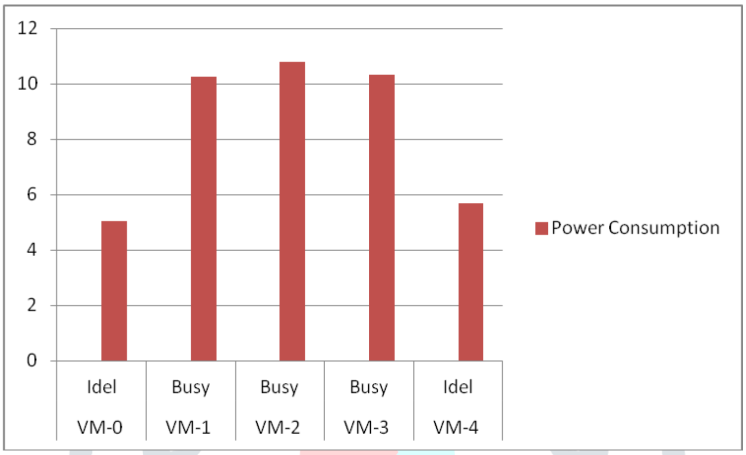

Figure: 4 Virtual Machine power consumption and status

V. CONCLUSION

This work proposed a modernist framework for power calculation of datacenters and servers. In Power calculation method we identify the Best and UnBest data center using their computational time which also used to differentiate Hot and Cold data centers of cloud using power calculation method. User requests placement techniques to find the best location of each request on the servers based on the typical data center configuration. List of Virtual machines are identified for the migration which consumes more powers then threshold value which is calculated by power model. This model helps to identify VMs for migration selection on other servers for load balancing. Our proposed work can help even more effective in future to reduce energy consumption in cloud computing.

VI. REFERENCES

[1]Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, Ivona Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility", Future Generation Computer Systems, vol. 25, pp. 599-616, 2009.
[2]Karthik Kumar, Yung-Hsiang Lu, "Cloud Computing For Mobile Users: Can Offloading Computation Save Energy", Published by the IEEE Computer Society, 2010.
[3]Nabil Sultan, "Cloud computing for education: A new dawn", International Journal of Information Management, vol. 30, pp. 109–116, 2010.
[4]Sukhwinder Kaur " Cloud Computing Applications In Multi-tenancy" Conference on Recent Innovations in Emerging Technology & Science, pp. 504 -507, April 6-7, 2018.
[5]Katarina Grolinger, Wilson A Higashino, Abhinav Tiwari, Miriam AM Capretz, "Data management in cloud environments: No SQL and New SQL data stores", Journal of Cloud Computing: Advances, Systems and Applications, vol. 2, no. 22, 2013.
[6]Fatma A. Omara, Sherif M. Khattab, Radhya Sahal, "Optimum Resource Allocation of Database in Cloud Computing", Egyptian Informatics Journal, vol. 15, pp.1–12, 2014.
[7]Maria Alejandra Rodriguez and Rajkumar Buyya,"Deadline Based Resource Provisioning and Scheduling Algorithm for Scientific Workflows on Clouds", IEEE Transactions On Cloud Computing, vol. 2, no. 2, pp. 222-235, 2014.
[8]Xingquan Zuo, Guoxiang Zhang, Wei Tan, "Self-Adaptive Learning PSO-Based Deadline Constrained Task Scheduling for Hybrid IaaS Cloud", IEEE Transactions on Automation Science And Engineering, vol. 11, no. 2, pp.564-573, 2014.

[9]Faiza Fakhar, Barkha Javed, Raihan ur Rasool, Owais Malik, Khurram Zulfiqar, “Software level green computing for large scale systems”, Journal of Cloud Computing: Advances, Systems and Applications, vol. 1, no. 4, 2012.

[10]Robert Basmadjian, Hermann De Meer, Ricardo Lent, Giovanni Giuliani, “Cloud computing and its interest in saving energy: the use case of a private cloud”, Journal of Cloud Computing: Advances, Systems and Applications, vol. 1, no. 5, 2012.

[11]Lingjia Tang, Mary Lou Soffa, Jason Mars, “Directly Characterizing Cross Core Interference through Contention Synthesis”, ACM 2011.

[12]Gaurav Dhiman, Giacomo Marchetti, Tajana Rosing, “vGreen: A System for Energy-Efficient Management of Virtual Machines”, ACM Transactions on Design Automation of Electronic Systems, Vol. 16, No. 1, 2010.

[13]Nazneen Taj, Dr. Anirban Basu, “International Journal of Engineering Research &Technology (IJERT)” ICIOT - 2016 Conference Proceedings, Volume 4, Issue 29, 2016.

[14]Soumya Ranjan Jena, V. Vijayaraja, Aditya Kumar Sahu, “Performance Evaluation of Energy Efficient Power Models for Digital Cloud" Indian Journal of Science and Technology, Vol 9(48), December 2016.

[15] Giuseppe Portaluri, Davide Adami, Andrea Gabbrielli “Power Consumption-Aware Virtual Machine Placement in Cloud Data Center" IEEE Transactions on Green Communications and Networking, Vol. 1, Dec. 2017.

[16]Ayan Banerjee, Tridib Mukherjee, Georgios Varsamopoulos, Sandeep K.S. Gupta,“Integrating cooling awareness with thermal aware workload placement for HPC data centers”, Sustainable Computing: Informatics and Systems, vol. 1, pp. 134–150, 2011.

[17] Demetrio Laganà, Michela Meo, “Reducing the Operational Cost of Cloud Data Centers through Renewable Energy” in Algorithms 11(10):145, September 2018.

[18]Gaurav Chadha, Scott Mahlke, Satish Narayanasamy, “When Less Is MOre (LIMO): Controlled Parallelism for Improved Efficiency”, ACM, 2012.

[19]Jordi Guitart, David Carrera, Vicenc Beltran, Jordi Torres, Eduard Ayguade, “Dynamic CPU provisioning for self-managed secure web applications in SMP hosting platforms”, Computer Networks, vol. 52,pp. 1390–1409, 2008.

[20]Anton Beloglazov, Jemal Abawajyb, Rajkumar Buyya, “Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing”, Future Generation Computer Systems, vol. 28, pp. 755–768, 2012.

[21]Nadjia Kara, Mbarka Soualhia, Fatna Belqasmi, Christian Azar, Roch Glitho, “Genetic-based Algorithms for Resource Management in Virtualized IVR Applications”, Journal of Cloud Computing, vol. 3, no.15, 2014.

[22]Antonio Corradi, Mario Fanelli, Luca Foschini,“VM consolidation: A real case based on OpenStack Cloud”, Future Generation Computer Systems, vol. 32, pp. 118–127, 2014.