Journal of Theoretical and Applied Information Technology
20th September 2014. Vol. 67 No.2
© 2005 - 2014 JATIT & LLS. All rights reserved·

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

FUTURE LOAD AWARE SERVICE BROKER POLICY FOR INFRASTRUCTURE REQUESTS IN CLOUD

[1]A.RADHAKRISHNAN, [2]V.KAVITHA

[1]Asstt Prof., Department of CSE, Anna University, Tirunelveli Region, India

[2]Assoc. Prof., Department of CSE, Anna University, Tirunelveli Region, India

E-mail: [1]arkrish77annauniv@gmail.in, [2]kavinayav@gmail.com

ABSTRACT

Cloud computing is a new computing paradigm, which offers resources for solving complex problem in fast and lower cost under pay per usage through internet. This feature is great boon to software companies to reduce their infrastructure setup and maintenance cost. IaaS service is one of the fundamental service models of cloud provider that offers entire computing environment as Virtual Machine (VM) to customers. The VM is created on cloud provider datacenter physical server. The provider deployed multiple datacenters in geographically different location to cater the needs of IaaS customers. The customers are much concerned about reducing their computation time and VM rental cost. The cloud service brokers are playing vital role in this regard, one of their responsibility is to direct the user request to appropriate datacenter. The selection of datacenter is a challenging task for service broker. Our proposed approach aims to enrich the intelligence of service broker during datacenter selection. Our novel algorithm makes the service broker to aware about the future load of every datacenter during request forwarding. It would facilitate the broker to route the IaaS request in right destination. The performance of our novel methodology is tested in Cloud Analyst tool. The result shows that our methodology reduces task time of customer applications compare to existing broker policies.

Keyword : *Virtual Machine (VM), Infrastructure as a Service (IaaS), Datacenters Classification Algorithm (DCA), Datacenter Load Aware Service broker Algorithm (DLASA), Neural Networks (NN).*

1. INTRODUCTION

Cloud Computing is a service oriented architecture, which is rendered through internet. Software industries are attracted by cloud services that minimizes their infrastructure setup and maintenance cost. Despite these potential benefits, customers are reluctant to do this business due to standing issues [1], [2]. Cloud provides its services in three different variants namely SaaS, PaaS and IaaS. SaaS service model contains built in applications along with required software and hardware, which are provided to customers as service. In PaaS, an application development environment is offered as service, using this facilities customer can build their own applications. In IaaS, a system instance is served as resource to customer. This instance could act as dedicated computer system to customer. The cloud provider has more than one datacenters to offers these services, which contains high configured servers and storages. The datacenters are placed in geographically different location to handle customer requests. The customers and providers are much interested parties to sustain their benefits. In IaaS service, the provider accountability is to provide reliable VM to customers, which avoid VM migration in middle of the computation and reduce processing time. The customer expectation is get completion of their task quickly, which minimizes the rental fee for IaaS service. The cloud service broker's responsibilities are high in this regard.

One of the activities of service broker is to receive the IaaS request from customers and directs the request to appropriate datacenter, whereas request could be processed. Choosing datacenter is tough task for service brokers. The selection process of datacenter is based on parameters such as response time of previous requests and current load of datacenters. Selecting unfeasible datacenter leads to VM migration very often inside the datacenter that ultimately lead to application span time. Our novel methodology refines the process of service broker datacenter selection policy. Through our approach the service brokers are aware about the near future load of each datacenter, based on this knowledge the selection will be done strategically.

Predict the near future load of datacenters computing resources can be tremendously useful for many activities. Resources instability in datacenter causes increase of application execution time that may lead to SLA slip. The resource availability predictor is framed by neural network with genetic algorithm, which forecasts the availability of datacenter resources to service broker during datacenter selection. Resource prediction is based on the resource monitoring. Each computing resource in datacenter consists of a data structure to keep track of its past load experience for prediction. The implementation and evaluation of our novel approach is done in Cloud Analyst tool.

2. LITERATURE SURVEY

Sergio Nesmachnow et al. [3] introduces a new formulation of task scheduling problem for multicore heterogeneous computational grid system in which the minimization of energy consumption metric is consider. Where, a meta broker agent receives all user tasks and schedule them on the available resources belongs to local provider. Antonio Cuomo et al. [4] proposed a procedure towards cloud@home, the major issues and challenges such as monitor, management and brokering of resources according to service level agreements are addressed by their framework. Yan Yao et al. [5] proposes network-aware virtual machine allocation algorithm for IaaS cloud environment. It minimizes the latency and communication cost between severs and number of virtual machines. Zhenzhong et al. [6] analyzes the multiple virtual machine migration problems and proposes scheduling method to reduces the migration time and accelerate the migration process. Thamariselvi Somasundaram et al. [7], [8] proposed a cloud resource broker method that facilitated with adaptive load balancing and elastic provisioning mechanisms. It handles the user application requests and balancing the load across the virtual instances. M. Ashok et.al. [9] proposed a methodology to identify the best resource for computation grid. The proposed solution focuses and extracts the trust of the resource with more accuracy. The Chao-Tung et al. [10] presented an environment for cloud system that provides a virtual environment to users. It immensely reduces the resource access time from cloud provider. Liang hu et al. [11] proposed the design and implementation of grid resource monitoring and prediction. In this approach, radial basis function is used for predicting grid resources. Kun Ago et al. [12] developed a framework for predicting task execution time that is used for task scheduling and resource allocation for forthcoming. Truong vinh Troung Duy et al. [13] illustrated the accurate resource load prediction approach for grid computing environment where back propagation neural network is proposed for grid resource availability prediction. Linlin Wu et al. [14] designed resource allocation algorithm for SaaS provider to minimize infrastructure cost and SLA violation, where quality of service parameters and infrastructure parameters are taken into account. Anton Beloglazov et al. [15] proposed green cloud architecture, which contains energy efficient resource allocation policies and scheduling algorithms considering QoS expectation and power usage characteristics of the device. Stillwell et al. [16] illustrates the heuristic resource allocation approach in homogeneous virtualized cluster environment, they assume each VM represents one computational job and described policies to allocate VM. Chase J.S et al. [17] proposed energy efficient management of resources in cloud hosting centers, where resource allocation is based on economic framework. This enables negotiation of Service Level Agreement (SLA) according to available budget and QoS requirements. Hemant S. Mahalee et al. [18] describes the performance of existing load balancing algorithm in CloudAnalyst tool, they concluded Throttled load balancing algorithm works more efficiently compared to other existing algorithms in terms of cost and datacenter processing time. Jasmin James et al. [19] proposed weighted active load balancing algorithm for cloud computing environment, where the VM assign a varying amount of the available processing power to the individual application services. Chaitali Uikey et al. [20] designed a trust model to calculate the rating of users, service provider by way of service broker. The broker chooses appropriate cloud service provider based on the request of user

3. CLOUD ARCHITECTURE WITH SERVICE BROKER

The general cloud architecture is represented in figure1. The architecture depicts the general structure of request received by cloud provider. The customers can avail this service from anywhere in the world through internet. The functional behavior of the major components in the architecture is described as follows.

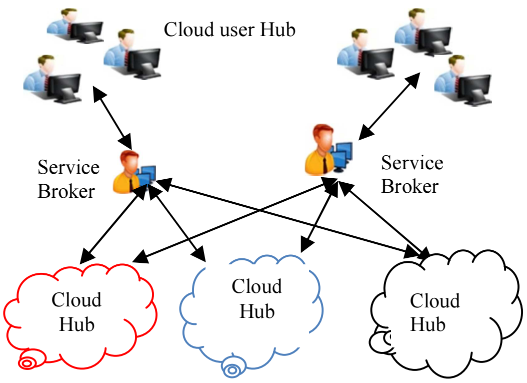

Figure 1: General Architecture of Cloud

User Base: The cloud user base consists of group of cloud customers, whose request are consolidate and forwarded to nearest service broker. The authentication processes are done as per the defined policies of cloud provider before requests are take in to account. The users must comply with Service Level Agreement executed with provider during their service tenure.

IaaS Service Broker: The service brokers are acting as intermediate entity between provider and user. The main role of this component is mapping IaaS requests into any one of the provider datacenter. This activity is done based on monitoring datacenters activities such as response time, and its work load.

IaaS Datacenters: Datacenter consists of high configuration servers with virtualization features. These servers are intended to cater to the need of aspirants. VMs are created on datacenter servers as per the request specification and render as service to customers. A provider own multiple datacenters in different location, they are able to migrate the computation among them.

NN Datacenters Load Prediction Process: This process is attached with each server in a datacenter hub. The main role of this process is to predict the near future load of datacenter server based on its past load history. To maintain the past load, a queue data structure is embedded in servers. The prediction mechanism is built by genetically weight optimized neural networks.

3.1 Request Handling Activates in Cloud

The sequence of activities take place in handing infrastructure service request is represented as sequence diagram in figure 2.

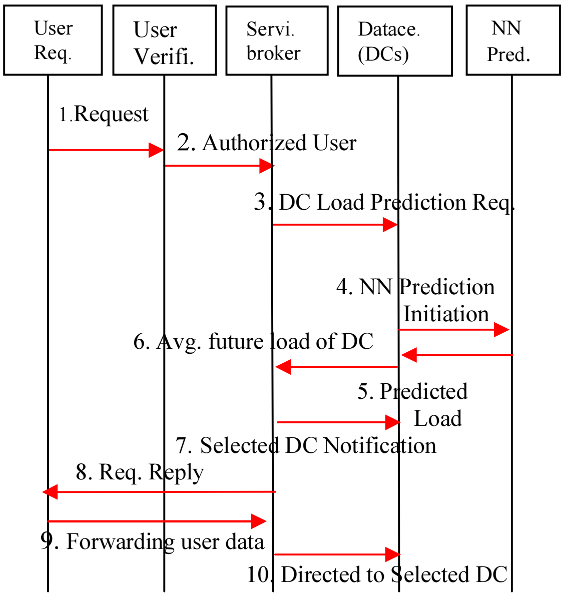

Figure 2: Sequence diagram of handling cloud customer Request

[1] The cloud customer send their IaaS request to nearest cloud service broker with resource specification.

[2] The cloud service broker initiate authentication process before forwarding the request to cloud datacenter.

[3] If request is authenticated then service broker trigger datacenters to initiate load prediction process.

[4] The NN model in each servers in datacenter, start predicting its near future load based on its past load history.

[5] Each datacenter calculates its average near future load and forwarded it to all service brokers.

[6] Service broker makes decision for choosing appropriate datacenter, considering future load as one of the prime parameter

[7] Service broker forward user data to selected datacenter for computation.

3.2 Classification of Datacenters

The datacenters contains number of high configuration servers to launch VM as per the request of IaaS customers. In our proposed methodology, the servers are affix with genetically weight optimized NN to assess the upcoming load. The assessed load of servers are used to categorize the datacenter (DC), the categorization is given in table 1.

Table 1: Categorization of Datacenters

Sl.No	Category of DC	Predicted Load (Average load of all servers in DC)
1	Heavy	>= 80%
2	Medium	Between >=70% and < 80%
3	Low	Between >=50% and < 70%
4	Very Low	< 50%

The measured load of each datacenter server is presented as value between zero and one. A datacenter load is taken as average predicted load value all of its physical servers. It leads a datacenter to fall in to any one the defined category. Heavy category of datacenters is normally never targeted by cloud service broker to forward requests.

4. ALGORITHMS

The pseudo code of novel algorithms DCA and DLASA are given below. These algorithms are embedded in datacenters and service brokers respectively. Based on load prediction, a datacenter is classified in to any of the category specified in table1. The DLASA algorithm is placed in cloud service brokers. It helps the service broker to strategically do their activity about datacenter selection for infrastructure requests.

4.1 Datacenters categorization Algorithm (DCA)

The objective of DCA algorithm is to classify the datacenters according to their near future load that helps service broker to make better decision. DCA takes a parameter L, which indicates how long past load is taken for prediction process. This value is determined by datacenter administration. The outcome of the algorithm is to split the datacenters according to their future load.

```
Algorithm: Datacenters categorization Algorithm
(DCA)
Input    : L {Indicates past L seconds host
              load for prediction }
Output   : Categorization of datacenters

Array:  Load-Pre [M][N]{M = No. of datacenter,
N = Number of servers}, DC-Cat[M], Heavy[M],
Medium[M], Low[M], Very Low[M].
Integer: Load-Ass = 0.
// Predict the load of each datacenter server by
using NN //
For each Datacenter$_i$ do {
  For each Server$_j$ do {
    Load-Pre[i][j] = Call genetically weight
                     Optimized NN (L)
    Load-Ass + = Load-Pre[i][j]
  }
  //Calculating average load of datacenter //
  DC-Cat[i]= Load-Ass /No. of Servers;
    Load-Ass ← 0
}
//Assign datacenters id to any one of the category
based on predicted load average of its servers //
For all DC-Cat do {
  Switch (DC-Cat[i]) {
    Case ‘Heavy’:Heavy[i] = DC-id
      break
    Case ‘Medium’:Medium[i] =  DC-id
      break
    Case ‘Low’:Low[i] = DC-id
      break
    Case‘Very Low’:Very Low[i]=  DC-id
      break
} }
Return(Heavy, Medium, Low, Very Low)
End
```

4.2 Datacenters Load Aware Service broker Algorithm (DLASA)

The main aim of DLASA algorithm is to identify a right datacenter for IaaS request. This algorithm using parameters such as category of datacenter based on future load, response time of previous requests and proximity of datacenters. The proximity gives the details about datacenters distance from user request base. The datacenters response time of previous requests are recorded in service brokers. This algorithm improves the intelligence of cloud broker to uphold benefit and trust on IaaS service in front of customers.

```
Algorithm: Datacenters Load Aware Service broker
Algorithm (DLASA)
Input     :  IaaS  Req,  Datacenter-Category,
Response Time, Proximity
Output    : Datacenter Identity

Begin
If (IaaS Req.User = ‘block List’) {
  Service forbidden
  break }
Else {
  Call the Procedure DCA
  //checking datacenters availability //
6. If Datacenter category Medium, Low, Very Low are NULL {
7. Hold the Request till any one of the category is set if it doesn't violates SLA }
8. Else {
//choosing suitable datacenter from datacenter category list//
9. If ('Very Low'! = NULL) {
10. Select a datacenter which proximity is comparatively near in 'Very Low category.
11.} Else {
12. If ('Low' != NULL) {
13. Select a Datacenter which response time and proximity is minimal from 'Low' category.
14. } If ('Medium' != NULL) {
16. Choose a datacenter in 'Medium' category which have less response history and proximity.
17. }}
18. Retrieve the identity of targeted Datacenter from the respective category for notification.
19.}} Return (Identity of datacenter)
20. End
```

5. EXPERIMENTAL SETUP

The experimental set up is made in two segment, in first segment, the neural network prediction system is constructed and assess its performance, the second segment consist of cloud environment creation to evaluate our novel methodology in IaaS service model. As cloud computing environment consist of loosely distributed systems, host load of a system is the most trustworthy information to judge system behavior. We have chosen "mystere10000.dat", a trace of workstation node for host load data set from http://people.cs.uchicago.edu. The experimental dataset is formed by two hundred samples of load taken in continuous order from "mystere10000.data". The transformation of selected dataset for neural network training is specified in table 2. In training, the number of input nodes, hidden nodes are five and single output node is taken.

Table 2: Data set format for Neural Network training

Output data	Input Feature m	…….	Input Feature 1
D(t)	D(t-1)	…..	D(t-m)
D(t-1)	D(t-2)	……	D(t-m-1)

The parameters such as MAE, R-square, CPU training time were used to measure performance of prediction. For increase the prediction performance the weight of neural network is optimized by Genetic Algorithm, in which two site crossovers is used and the chromosomes with slightest error are taken for the Genetic Operations. These factors are applied on three neural network models namely Back Propagation Neural Networks (BPNN), Elman Neural Networks (ELNN) and Jordon Neural Networks (JNN) to measure their prediction performance.

The IaaS cloud environment setup is fabricated in CloudAnalyst toolkit. The CloudAnalyst toolkit supports both system and behavior modeling of cloud system components such as datacenters, VMs and service broker policies. CloudAnalyst supports service broker policy in two levels namely Service Proximity based Routing (SPR), Performance Monitoring Routing (PMR). In SPR, the service broker route the user traffic to the closet datacenter based on transmission latency. In PMR, the service broker monitors all the datacenters, among those which one given the best response to the user past request that would be targeted for current request. We have compared the performance of our proposed cloud service broker policy against SPR and PMR. The following setup is made in Cloud Analyst tool as represented in table 3.

Table 3: CloudAnalyst Simulation setup

SIMULATION SETUP	
Number of User Base	5
Number of Datacenter	3
Datacenter OS	Linux and windows
Physical Hardware unit	2
Number of Processors	4
Number VM	15
Service Broker policy	(i) SPR (ii) PMR
VM load Balancer	Round-Robin
Request per user / hour	60
Data size per request (Bytes)	100
Internet Characteristics	Default

The simulation is executed with existing service broker policies such as SPR, PMR and internet characteristics as specified in table 3, next proposed service broker policy DLASA is incorporated in the tool since CloudAnalyst is open source and developed in JAVA language. The result of existing service broker policies and our novel methodologies is analyzed in next section.

6. RESULTS AND DISCUSSION

The prediction performance of BPNN, ELNN and JNN are depicted in Figure 3. It represents the value of evaluation parameters such as MAE, R-Square and CPU time taken. The MAE gives the accuracy of prediction, R-Square statistical measure represents, how much successful fitting is attained between targeted and predicted values. The CPU training time is measured in milliseconds (ms), which denote the time taken for complete the task of prediction.

As per the result, JNN R-Square value is closer to one compare to remaining neural network model as well as its CPU time taken also very less against others. Despite the R-Square and MAE value of ELNN is closer to JNN, the CPU time taken is high compare to JNN. The BPNN performance is poor compare to ELNN and JNN. Based on evaluation parametrs result, the JNN has chosen for prdict the near future load of datacenters servers.

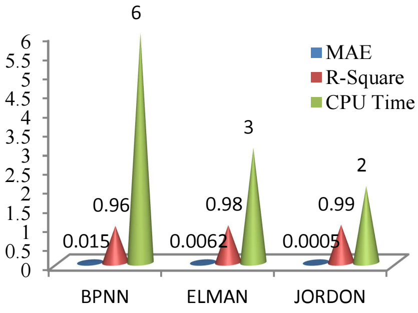

Figure 3: Genetically weight optimized NN Prediction

The Cloud Analyst simulation screen contains map of the world, which divides into six regions. It facilitates to place datacenters and user hub at any region and set their characteristics. As per the experimental setup, three datacenters and five user bases are deployed in different region. The simulation is first run with existing cloud broker policies SPR and PMR after that it run with our novel cloud broker policy DLASA. The performance is depicted in figure 4. It shows that the response time of five user bases using SPR, PMR and DLASA. The proposed service broker policy DLASA significantly reduces response time of customer requests

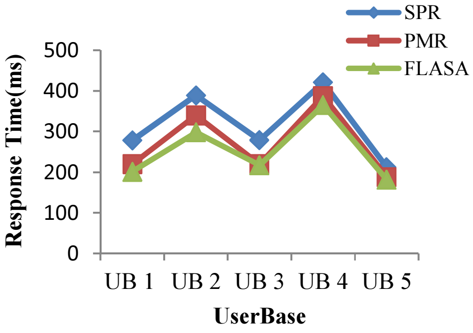

Figure 4: User Base Response Time Comparison

The existing service broker policy PMR is closer to DLASA, however most of the user base get less response time compare to PMR. This impact is reflected in datacenters processing time too. The processing time of datacenters is significantly reduced by DLASA that is presented in table 4.

Table 4: Data Centers Processing Time comparison (Measured in milliseconds)

Datacenters	SPR	PMR	FLASA
DC-A	7896	6018	5897
DC-B	8972	7734	7535
DC-C	8992	7561	7419

7. CONCLUSION

Cloud computing is an internet based computing technology, where required resources are provided in rented basis to customers. Moving computation and data in cloud datacenters provides great conveniences to cloud users. One of the fundamental cloud services is IaaS, where machine instances are provided as resource to customers. The customers are availing this service through cloud service brokers. The primary task of service broker is route customers request to suitable datacenter. The provider deployed multiple datacenters in geographically different location to serve customers. Before routing, the service broker considers the parameters such as proximity, response time and current load of datacenters to forward the request. In this paper a novel service broker policy is proposed, which uses the near future load of datacenters as one of the key parameter to select a datacenter. The future load prediction is done by genetically weight optimized Jordan neural network. This methodology is stuffed in our novel algorithm DLASA. The performance of proposed service broker policy is tested in Cloud Analyst tool. The result shows that the proposed methodology significantly minimizes the response time of cloud customer requests and drastically reduces the processing time of datacenters. These impacts amplify the benefits of IaaS customers and IaaS service provider.

REFRENCES:

[1] S. Subashini , V. Kavitha , " A Survey on security issues in service delivery models of cloud computing", *Journal of Network and Computer Applications,* Vol. 34, Issue 1, 2011.

[2] Luis M. Vaquero, Luis Rodero Merino, Daniel Moran. "Locking the sky: a survey on IaaS Cloud Security", *Journal of Computing,* Vol. 91, Issue1, January 2011, pp 93-118.

[3] Sergio Nesmachnow, Bernabe Dorronsoro, Johnaton E. Pecero, Pascal Bouvry, "Energy-Aware Scheduling on Multiware Heterogeneous Grid Computing System" , *Journal of Grid Computing,* Vol.11, Issue 4, December 2013, pp 653-680.

[4] Antonio Cuomo, Giuseppe Di Modica, Salvatore Distefano, Antonio Puliafito, Orazio Tomarchio, Salvantore Venticinque, Umberto villano, "An-SLA based Broker for cloud infrastructure", *Journal of Grid Computing,* Vol. 11, Issue 1, March 2013, pp 1-25.

[5] Yan Yao, Jian Cao, Hingluli, "A Network Aware Virtual machine Allocation in Cloud Datacenters", *Journal of Network and Parallel computing,* Vol. 8147, 2013, pp 71-82.

[6] Zhenzhong, Limin Xiao, Xianchu Chen, Junjie Peng, "A Scheduling Method for Multiple Virtual Machine Migration in Cloud", *Journal of Network and Parallel Computing,* Vol. 8147, 2013, pp 130-142.

[7] Tamariselvi Somasundaram, Kannan Govindarajan, M. R. Rajagopal , S. Madhusudana Rao, " A Broker based Architecture for Adaptive Load Balancing and Elastic Resource Provisioning in Multi-tenant based Cloud Environment", *International conference on Advances in Intelligent system and Computing,* Vol. 174, 2012, pp561-573.

[8] Tamariselvi Somasundaram, Kannan Govindarajan, UshaKiruthika, Rajkumar Buyya, "Sematic–enabled CARE Resource Broker for managing grid and cloud environment", *Journal of Super Computing,* January 2013.

[9] Ashok, S. Sathiyan, "A Parameterized Framework of Trust Computation for Grid Resource Broker", *Journal of Trends in Computing and Communication System",* Vol. 269, 2012, pp 178-181,.

[10] Chao-Tung Yang, Bo-Han Chen, Wei-Sheng Chen, "On Implementation of a KVM IaaS with monitoring system on Cloud Environment", *Journal of Communication and Networking,*Vol. 265, 2012, pp 300-3008.

[11] Liang Hu, Xi-Long Che, Si-Qing Zheng, "Online System for Grid Resource Monitoring and Machine Learning-Based Prediction", *IEEE Transactions on Parallel and Distributed Systems,* Vol.23, No. 1, January 2012.

[12] Kun Gao, Qin Wang and Linfeng Xi, "Reduct Algorithm Based Execution Times Prediction in Knowledge Discovery Cloud Computing Environment", *International Arab Journal of Information Technology,"* Vol. 11, No 3, May 2014.

[13] Truong Vinh Truong Duy, Yukinori Sato, Yasushi Inoguchi, "Improving Accuracy of Host load predictions on Computing grids by Artificial Neural Network", *International journal Parallel, Emergent, Distributed system*, Volume 26, Issue 4, 2011, pp 275-290, .

[14] Linlin Wu, Saurabh Kumar Garg and Rajkumar Buyya, "SLA–Based Resource Allocation for SaaS in Cloud Computing Environments," *11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing,* 2011.

[15] Anton Beloglazov, Jemal Abawajy , Rajkumar buyya, "Energy aware resource allocation heuristics for efficient management of datacenters for cloud computing", *Future Generation computer System,* Vol. 28, Issue 5, 2012 .

[16] M. Stillwell, D. Schanzenbach , F. Viivien and H. Casanova, "Resource allocation using virtual clusters", *9th IEEE Symposium on Cluster computing and Grid,* 2009.

[17] J.S Chase , D.C. Anderson, P.N. Thakar , A.M. Vahdat and R.P. Doyle, "Managing Energy and server resources in hosting centers", *18th ACM Symposium on Operating system principles,* New York 2011.

[18] S. Hemant .R. Mahalee, Parag . Kaveri, Vinay Chavan, "Load Balancing On Cloud Data Centers", *International Journal of Advanced Research in Computer Science and Software Engineering,* Vol. 3, Page 1, January 2013.

[19] Jasmin James, Bhupendra Verma, "Efficient VM load Balancing algorithm for a Cloud Computing Environment", *International Journal of Computer Science and Engineering,* Vol. 4, September 2012, pp 09.

[20] Chaitalai Uikey, D.S. Bhilare, "A Broker Based Trust Model for Cloud Computing Environment", *International Journal of Emerging Technology and Advanced Engineering,* Vol. 3, Issue 11, November 2013.