

Energy-Aware Load Balanced Scheduler for Cloud Computing to Improve Quality of Service

N. R. Ram Mohan^{1*}, E. Baburaj²

¹ Computer and Information Technology, Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India.

² Dept of Computer Science and Engineering, Sun College of Engineering and Technology, Nagercoil, Tamilnadu, India.

* Corresponding author. Email: nrrammohan@yahoo.co.in.

Manuscript submitted June 7, 2014; accepted June 8, 2015.

doi: 10.17706/jsw.10.7.869-880

Abstract: Cloud computing paradigm has progressed as extremely ascendable services with increasing power on computation, massive storage possibility and the resources being obtained as service using cloud environment by providing guarantee to the Service Level. However, the subscriber requirement has increased to an extent that necessitates a big active platform for balancing the load though the available resources are shared according to the availability. With the advent of powerful network processors, the cloud computing paradigm also give rise to load balancing problem which has to be solved in an optimal manner in order to avoid deadlock and enhance resource effectiveness. To address the issue of energy conservation in cloud computing paradigm, the use of progressive traffic data from data centers applied a service invocation forecast model. The collaborative data possession scheme applied responsive pattern using the Homomorphic verifiable format and hierarchy was performed using hash index model but the disadvantage of the model was that the match index structure did not matched properly using cluster. However enforcing policies in cloud computing paradigm is challenging because of the different slabs of power tariffs and requirements made to the servers affect the decisions, whether the loads to be pushed in or out of a cluster affecting the overall energy utilization. To address the issue of balancing the load and optimizing the bandwidth and energy utilization, an Energy-aware Load Balanced Scheduler (ELBS) for cloud computing to improve Quality of Service is presented.. The effectiveness of the proposed ELBS model is illustrated by theoretical analysis with Virtual Machine (VM) energy-efficient cloud data centers. Performance metric for evaluation of ELBS model is measured in terms of computational complex for energy utilization, performance (in terms of throughput), bandwidth utilization rate, cloud computation cost, and response time to service invocation, load balance to improve QoS and clustering quality.

Key words: Service level agreement, energy-aware load balanced scheduler, virtual machine, cloud computing paradigm, bandwidth utilization, query messaging, data centers.

1. State of Art

Cloud computing paradigm is changing the lifestyles of the clients and extensively changes the way people analyze the information. With the configuration advancement in the terminal devices, the cloud offers insights into differing platforms owned by different users to operate. The next saga of cloud user devices provides insights not only for continuum in operation, but also optimal utilization of information. With these environment in hand the computation model of the neighborhood nodes, storage of information

and message exchanges becomes efficient. The cloud computing paradigm provides an efficient method that comprises of both the infrastructure pattern in a secured way and convenient model by minimizing the operation cost. On the other hand, the cloud computing paradigm heavily depends on the data centers as their primary means that also include computing infrastructure. The Cooperative provable data possession method envisages the technique of homomorphic verifiable responses and hash index hierarchy. Both the homomorphic verifiable responses and hash index hierarchy is still considered as the challenging problem during the process involved in scheduling with the data size irrelevant to the size of blocks as explained in [1].

The scheduling algorithm based on the factor cost as designed in [2] made effective task mapping that evaluated the resource cost of resource and computation cost and at the same time recovered the computation ratio by grouping the tasks of the user tasks on the basis of the cloud resources availability. But, however the algorithm enhancement did not paved way for task scheduling on an independent manner in the cloud computing paradigm. Moreover with the consideration of both requirements of the user and properties required to provide the infrastructure pattern, the cloud scheduler did not concentrated on extending the trustworthiness of certain other certain user requirements.

Trustworthy scheduling algorithm as designed in [3] failed in measuring the constructing up resource's RCoT and the measurement involved during the integrity. The framework of decentralized accountability information model keeps the precise handling of the user integrated in the cloud. In specific, the object-centered paradigm as illustrated in [4] enabled the mechanism of logging integrated with the data and policies of the users' but failed to provide the trustworthiness and data integrity of the JRE and the JAR authentication.

The most significant model designed for auditing the data stored in cloud is the third party auditor model where the data was verified in a public manner and also allowed the third party auditor to audit the storage of data in cloud data without the necessity of the need of the users' time, feasibility or resources. On the other hand, the auditing performed in external manner referred to as the third party auditor (TPA), ensured the data integrity on behalf of the cloud user for storage of data in the cloud. The storage of data in cloud as demonstrated in [3] utilized public key mechanism on the basis of authentication performed using homomorphic mechanism in addition to the privacy preserving public auditing.

The job of performing the third party auditor (TPA) as demonstrated in [1] provided with the data integrity for the purpose of storage of data in a dynamic manner. The objective of TPA lies behind the elimination of the integration of the client with the help of auditing the storage of data in the cloud. Moreover, TPA achieved maximum cloud scalability where the verifiability was achieved in a public manner with the advantage of not requiring the TPA to audit the storage of data in cloud without compromising the time and resources of the users'.

In cloud computing paradigm, the allocation of resources is based on the elasticity requirements and remains as the key mechanisms when compared to the existing multiprocessor task allocation. Resources achieving scalability as demonstrated in [5] made optimal resource allocation as an important issue but failed in providing relaxation, task tardiness and providing bulk discount pricing. The automation process of allocation of resources during parallel processing as demonstrated in [4] does not presume the data statistics availability and semantics involved during the application of data but resulted n the maximum tradeoff between benefits being provided and overheads obtained. A mechanism of resource sharing in an optimal manner called as the Federated cloud maximized the factor of salability. Resource allocation in the cloud computing paradigm as illustrated in [4] involved complexity during the process of resource allocation and as a result, method based on the agent failed in incorporating the model involved in the protocol and the system was also tested with the help of JADE, a programming language.

The users evaluate the requirements of their resources including the required hard disk space, memory constraints, CPU, network bandwidth, and integrate them all in a combined manner into virtual machines (VMs) by mapping it to the substantial resources. One of the important issues to be solved in the cloud computing paradigm is the optimal allocation of resources. Resource allocation in cloud is one of the major problems to be solved in cloud computing. Several researchers have designed different techniques for it. Wavelength-division multiplexing (WDM) using Ethernet as illustrated in [2] paved the means for cost-effectiveness that provide transfer of data involving data-intensive applications. However, neither the conventional methods included precise paths between the pairs of source destination nor the conventional application-level approaches. But the conventional approach heavily depended on the resources being computed but includes the pattern of basis model of utilization of power for being taken into granted.

The method Dynamic Power Management (DPM) resulted in the maximization of the savings when compared to the average energy utilization in cloud computing systems. But the method DPM related highly to the Dynamic Voltage and Frequency Scaling (DVFS) technology that provides the optimal performance of hardware and utilization of energy in order balance the corresponding tasks of the workload. A significant level of energy was conserved as illustrated in [1] by shifting the virtual machines (VM) deployed on underutilized equipment to increased machines and underutilized machines. The design of virtual machines at the other end aimed to provide a policy for energy-efficient cloud data centers. It made use of progressive traffic data from data centers by using a service invocation forecast model.

The capacity of electric power in each area as demonstrated in [4] assumed a cloud computing environment in which both the parametric factors such as the ability to be processed and bandwidth utilization of network were simultaneously allocated. But the capacity of power for integrating the requirements of several disciplines and applications failed to switch between servers and bandwidth to switch from the sleep mode to the operating mode. These types of services were provided by the Cloud computing services provider. The resources for them were issued on-demand in order to provide the requirement of the Service Level Agreement (SLA). The framework of Service Level Agreements (SLA) differentiated between the workloads under bounded latency requirements for the purpose of savings of cost on geographical load balancing.

The execution of workload as demonstrated in [6] did not follow an analysis on the basis of probabilistic for cost saving in demand response market. SLA in cloud computing is a precise model the main constraint the Quality of Service (QoS) due to the fact that the method provided means for abstract type of services correlated to provide a specific goal. Moreover, the technologies involved in virtualization shifted the virtual machines to physical resources on the basis of changes in load in order to accomplish the load balance for the entire system in a secure manner.

The outsourcing mechanism involving a secured pattern of linear equations (LE) as demonstrated in [3] provided the LU decomposition. The Statistical Process Control (SPC) charts as demonstrated in [6] evaluated and measured the performance anomalies and differential profiling to identify their root causes. With the task automation, the model failed to expand the scope of automation, based on certain detailed analysis of profiling data.

Security-mediator (SEM) as demonstrated in [2] provided verification for metadata on data being outsourced for data owners and provided anonymity protection mechanism. The measure of fairness in congested environment as demonstrated in [7] provided multiple resource types, which were simultaneously allocated simultaneously to each service invocation. As a result, congestion control method provided fair allocation of resources between users in congested environment.

The decentralized types of resource allocation as illustrated in [4], used market-based heuristics did not enrich in CloudSim where it failed in a systematic manner while adding these modifications. A method that

provided performance tradeoff as illustrated in [4] shared data confidentiality on cloud servers using Hierarchical Identity Based Encryption (HIBE) system and the cipher text-policy attribute-based encryption (CP-ABE) system. To provide fine-grained and scalability during the data access control, the patient's health records were encrypted as illustrated in [7]. On overlooking the above given techniques and methods,

Based on the aforementioned techniques and methods, in this work, an Energy-aware Load Balanced Scheduler (ELBS) for cloud computing to improve Quality of Service is presented. The ELBS algorithm is deployed on the basis of clustering where cluster heads are elected in an intermittent manner to maintain optimal level of balance between the respective clusters to minimizing the overall communication cost in cloud computing paradigm. The primary focus and objective of ELBS is to minimize the response time of the tasks by workload distribution. The second objective of ELBS is to enhance the tradeoff performance by distributing load system in cloud computing paradigm. An energy aware load balanced scheduler optimizes the utilization of energy utilization of the data center by load balancing of traffic surrounded by the data center. An effective load distribution in the cloud computing paradigm improves the Quality of Service (QoS) by minimizing delays during the communication and loss of information due to congestion in cloud computing paradigm.

The organization of the paper is given as follows. In Section 2, different types of works with their disadvantages are presented. In Section 3, a detailed study of design and operational considerations for sketching the motivation is presented. Section 4 provides detailed analyzes of the parameter for the overall system with the help of experiments. The paper concludes with the concluding remarks in Section 5.

2. Energy-Aware Load Balanced Scheduler in Cloud Computing Paradigm

In this section, we present our energy-aware load balanced scheduler in cloud computing paradigm to improve the Quality of Service using ELBS algorithm. Two methods are designed for constructing the ELBS namely, clustering server family where the servers are grouped and progressive query messaging to achieve the goal of scheduled load using clusters. The overall architecture diagram of the Energy-aware Load Balanced Scheduler is illustrated in Fig. 1.

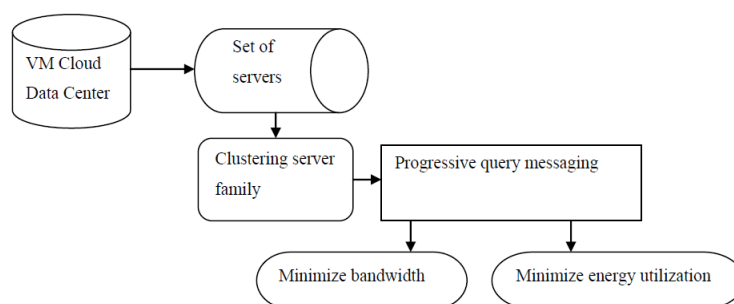


Fig. 1. Framework of ELBS model.

The ELBS model in addition to the VM cloud data center executes for every received data from the cloud computing data center. A family of servers is in a way connected to the VM data center with the maximum bandwidth available and is designed in such a way that at least one of the servers meet the demands of the users. Moreover, the ELBS algorithm equalizes the loads with progressive query messaging with minimum communications between the cloud computing paradigms. The ELBS algorithm considers the cluster head whenever the imbalances occur with respect to a load threshold. Energy-aware Load balanced Scheduler finally achieves minimal bandwidth and energy utilization through repeated operations.

To make the work self-contained, the background behind the design consideration of ELBS and the framework of ELBS is given in 2.1. Section 2.2 formulates our model using the progressive query messaging.

To achieve the design goal, the process involved during the congestion in the server is explained in 2.3. The detailed process involved is discussed in the forthcoming section.

2.1. Energy-Aware Load Balanced Scheduler Mathematical Instruction

The process involved in Energy-aware Load Balanced Scheduler (ELBS) is to identify the most specific family to share the load evenly to avoid imbalances. The energy-aware load balanced scheduler for cloud computing paradigm is designed in such a way that optimization of utilization of energy in the corresponding data centers is achieved. The ELBS model equally evaluates the users' communicational demands to that of the computing requirements.

In the VM cloud data center, the server comprises of 'N' frame and stored in the component 'Comp'. Many frames comprise of set of components and use the family of servers with the highest bandwidth available. The principle behind ELBS is that the ELBS initially identifies the component 'Comp' in such a way that

$$Band(Comp_i) = MAX (Band(Comp_i)) \quad (1)$$

where, 'Band' specifies the bandwidth available for a component $Comp_i$ evaluated for a per-server basis. For a component $Comp_i \in Comp$ the bandwidth available is given as,

$$Band(Comp_i) = \frac{Tranc_i - transrate}{(\sum c_i)} \quad (2)$$

where, $Tranc_i$ denotes the transmission capacity for a component c_i , measured as a sum of highest transmission speeds of all links with respect to a component 'c' to the cloud computing paradigm, $transrate$ is the effective transmission rate, and $\sum c_i$ is the sum of the servers in the component. In a similar manner a frame is identified with the highest bandwidth available, ELBS evaluates a frame $frame_i \in F$ such that

$$Bandframe(f_i) = MAX(Bandframe(f)) \quad (3)$$

Bandframe is the bandwidth available bandwidth for a frame f_i evaluated for a per server basis cloud computing paradigm. For a component $c_i \in F$ the available bandwidth computed as

$$\begin{aligned} Bandf_i(t) &= \frac{1}{TI} \int_0^{0+TI} \frac{Tranf_i - transrate(t)dt}{\sum f_i} \\ &= \frac{1}{Sum\ of\ f_i} (Tranf_i - \frac{1}{TI} \int_0^{0+TI} transrate(t)dt) \end{aligned} \quad (4)$$

where, $Tranf_i$ represents the transmission capacity for a frame 'i', which is obtained as a sum of highest transmission speeds for all the links connecting to a frame 'i' of cloud family, $transrate(t)$ represents the transmission rate of the specified user and $\sum f_i$ denotes the number of servers in a frame in a cloud computing paradigm.

2.2. Progressive Query Messaging

As soon as the components and frame are identified with the highest possible bandwidth available in ELBS model, a high energy-aware node is elected as the cluster head as energy being the most critical resources in ad hoc networks. In ELBS model, the advantage of the cluster head is that it consumes less energy than the other. The task of cluster head includes the coordination between other members in the cluster family, regular maintenance of cluster and balancing the load. In order to minimize the frequent changes in the selection of cluster heads in cloud computing paradigm, the information that are being updated are selected followed by cluster head evolution.

As illustrated in the Fig. 2, the major role of a cluster head involved in the ELBS model is to maintain the load balancing using cluster. The task of the cluster head is to collect the data center information in a periodical manner consisting of the energy and load and the corresponding values are stored in the member table. Whenever a server reaches the overload from users, a message is transmitted to its cluster head. The secondary server then makes a decision with respect to its associate table and selects the nodes with the minimum load and the energy accompanied by a response message sent to the corresponding users. As soon as the new server enters into the cluster family, the associate table is updated accordingly. The load balancing algorithm for ELBS is explained as below:

Begin

Initialize

Input: Time Interval 'TI', $Tranf_i$ is the transmission capacity of a frame 'i', $\sum f_i$ is the number of servers initiated in VM data center

Output: Minimal bandwidth and Energy Utilization

Assign VM cloud Data Center

If $TI > Tranf_i$ then

Condition=False

End if

For 'n' servers

Evolve cluster head from server group

Evaluate associate table with values, energy and load

If (overhead occurs) then

Message transmitted to cluster head

Else

Associate table send response to users with least load and energy capacity

End If

End For

End

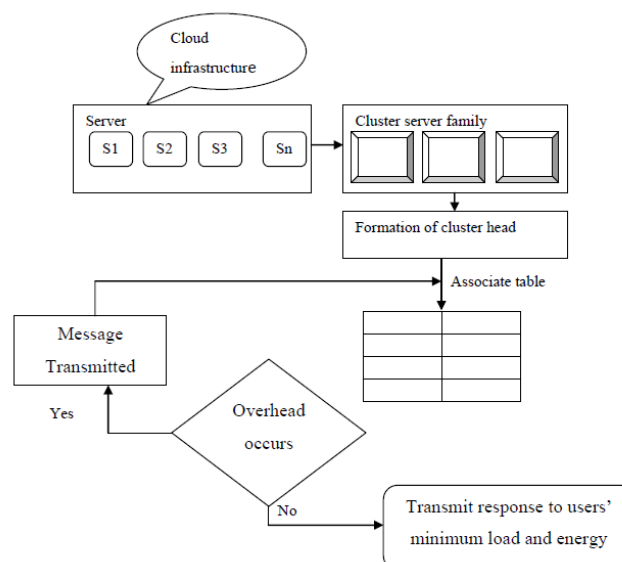


Fig. 2. Illustration of Load Balance using cluster.

The above pseudo code explains the process involved in ELBS model. The associate tables are maintained by the cluster head server in order to control the load occurring from their members. Concurrently, each

node in the cloud computing paradigm transmits a message followed by the energy and load values to the corresponding cluster head which updates its table in an intermittent manner. The thresholds for ELBS model are designed for each server in such a way that the server carry out the process of transmission with minimum energy. Correspondingly each server checks the load and energy and compares them with the two thresholds in an intermittent manner. If one of the two thresholds is attained then the server sends a message to its cluster head. As a result, the ELBS communicates with its associate table and selects the one which has the minimum load and the energy capacity. If one certain node is identified then the cluster head sends a positive response to the server implying the address of the new server that then receives the additional load.

2.3. Preventing Congestion in Server Using ELBS Model

As soon as the process of clustering and query message is performed, the concluding goal of the ELBS model is to maintain load balance and prevent congestion occurring in the server. To obtain a more precise measure for avoiding congestion in the server, ELBS model equates the measures of the bandwidth available with respect to the components $Band(Comp_i)$ and frame $Bandf_i(t)$ with the component with respect to the size of the row.

$$Row(time) = 1 - \frac{1}{TI} \int_0^{0+TI} (e^{-\frac{(r(t)-1)}{Row_{max}}}) dt \quad (5)$$

where, $Row(time)$ evaluates the instantaneous occupancy of the row measured during the time interval 'time', Row_{max} is the maximum allowed size of the row. Moreover, the ELBS model also establishes the speed of load balancing with the increasing congestion.

3. Experimental Results of ELBS Model

The energy-aware load balanced scheduler model for improving the quality of service is measured against the Virtual Machine (VM) for energy-efficient cloud data centers. For experimental purposes, set of parameters are taken for evaluation and deployed in JAVA CloudSim simulator. The ELBS model is compared against the Energy conservation in cloud infrastructures with Service Invocation Forecast (SIF) model and Homomorphic Verifiable responses Hash Index Hierarchy (HIH) model in terms of computational complex for energy utilization, performance (in terms of throughput), bandwidth utilization rate, cloud computation cost, and response time to service invocation, load balance to improve QoS and clustering quality.

The average amount of energy used for the query processing in cloud computing paradigm in the computation complex for energy utilization. The computation complex for energy utilization of ELBS model existing system is measured in terms of Joules (J) with bandwidth utilization performed to measure the usage of processing resources for balancing the load of each factor, measured in terms of Kilo bits per second (Kbps) whereas the performance (in terms of throughput) for ELBS model is the effective result obtained on the overall system in cloud computing paradigm measured in terms of percentage (%).

The response time to service invocation for ELBS model is evaluated that measures the average amount of time consumed to respond to the requirement send of the clients which is measured in terms of seconds (sec). The load balance factor for ELBS model for multiple servers evaluates the maximal throughput used to avoid the overload and increases the reliability which is measured in terms of Mega Bytes (MB). Finally, the clustering quality for ELBS model is defined with effective result on grouping of similar servers to improve the query processing result which is measured in terms of percentage (%).

4. Performance Result of Adaptive Load Balancing

The Energy-aware Load Balanced Scheduler (ELBS) model is compared against the existing Energy conservation in cloud infrastructures with Service invocation Forecast (SIF) model and Homomorphic Verifiable responses Hash Index Hierarchy (HIH) model. The table given below and graph describes the ELBS model when compared with existing system.

Table 1. Tabulation of Energy Utilization Using different Models

VM counts	Computational Complex for Energy Utilization (J)		
	SIF model	HIH model	ELBS model
3	6.26	4.58	3.15
6	8.15	6.18	5.07
9	9.18	7.22	6.25
12	11.89	10.26	8.89
15	13.25	11.23	8.55
18	17.46	15.95	13.45
21	20.91	18.14	15.25

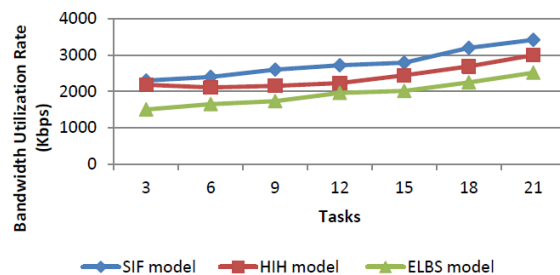
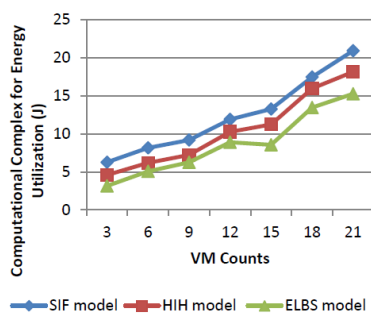


Fig. 3. Measure of computational complex for energy utilization. Fig. 4. Measure of bandwidth utilization rate.

Table 1 and Fig. 3 illustrate the measure of energy utilization on the basis of virtual machine counts (VM). The utilization of energy is increased step by step with the increase in the count of virtual machine. With the increase in the count of virtual machines, the energy utilization of ELBS is reduced to approximately 35 – 44 % when compared with the SIF [1] model. This is because the ELBS model evaluates the load and its energy and checks them with two thresholds in the associate table. With the introduction of associate table during the processing of query, the ELBS model minimizes the energy utilization which is dropped to 18 – 28 % when compared with the HIH model [7].

Table 2. Tabulation for Bandwidth Utilization Rate

Tasks	Bandwidth Utilization Rate (Kbps)		
	SIF model	HIH model	ELBS model
3	2300	2185	1505
6	2400	2110	1650
9	2600	2155	1735
12	2720	2230	1960
15	2790	2445	2015
18	3200	2690	2250
21	3420	3000	2515

The bandwidth utilization rate of the ELBS model is compared with the SIF [1] model and HIH model. With the increase in the number of task, bandwidth utilization rate in the cloud computing paradigm is minimized to 13 – 18 % in ELBS model than when compared to the SIF model. The bandwidth utilization

rate of ELBS model is also compared with the HIH model, where the result obtained is examined. The bandwidth of a frame f_i in ELBS model is measured in terms of single server which minimizes the bandwidth utilization rate to 3 – 12 % when compared with the HIH model.

Table 3. Tabulation for Performance (in Terms of Throughput)

Users	Performance (in terms of throughput) (%)		
	SIF model	HIH model	ELBS model
30	65	54	79
60	63	50	78
90	62	52	77
120	61	58	77
150	60	51	76
180	60	53	74
210	59	48	73

Table 3 describes the performance value in terms of throughput for the SIF model, HIH model and ELBS model. The users taken for the measure ranges from 30, 60, 90... 210. As the number of user increases, the performance is also improved using ELBS model. The experimental result are plotted in graph and depicted in Fig. 5.

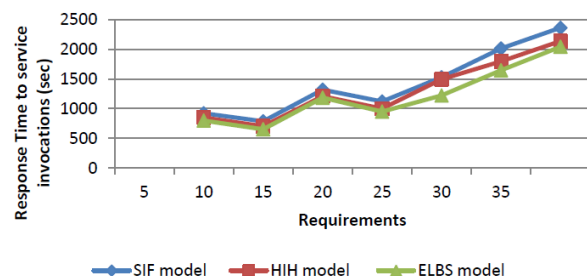
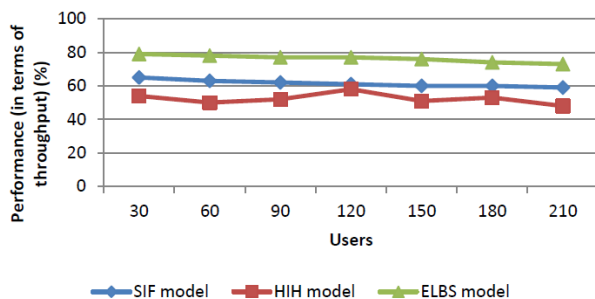


Fig. 5. Measure of performance (in terms of throughput). Fig. 6. Measure of response time to service invocations.

Fig. 5 describes the performance of the ELBS model with the existing system. The figure illustrates that the performance is improved using ELBS model as the ELBS scales the measures of the available bandwidth $Band\ c_i(t)$ and $Bf_i(t)$ within the component with reference to the size of the row, the performance is improved rapidly. The performance of ELBS model is improved by 10 – 15 % when compared to the SIF [1] model and 15 – 20 % when compared with the HIH model.

Table 4. Tabulation of Response Time to Service Invocations

Requirements	Response Time to service invocations (sec)		
	SIF model	HIH model	ELBS model
5	923	851	799
10	787	704	652
15	1321	1210	1185
20	1121	1002	950
25	1530	1495	1223
30	2016	1799	1647
35	2365	2140	2045

Table 4 and Fig. 6 describe the response time to service invocation using the SIP model, HIH model and ELBS model. The requirement considered for the experimental purpose are 5, 10, 15 ,... upto 35 requirements. The response time to service the requirement is better using ELBS model than the SIP [9] model and HIH [7] model as the time taken to respond using ELBS model for a frame 'i', is evaluated using $Tranf_i$, where the maximum response speeds of all links connecting a frame 'i' in a cloud family is also analyzed. The ELBS model uses the association table to reduce the response time drastically when compared with the SIP model [9]. The response time using ELBS model is 12 – 17 % when compared to the SIP model and 4 – 10% minimized when compared to the HIH model.

Table 5. Tabulation of Cloud Computation Cost Using Load Balance

Server Number	Cloud Computation Cost using Load Balance (MB)		
	SIF model	HIH model	ELBS model
Server 1	65	60	55
Server 2	70	63	58
Server 3	72	65	62
Server 4	75	68	67
Server 5	82	70	68
Server 6	85	72	70
Server 7	88	75	71

The Table 5 describes the Cloud Computation Cost using Load Balance factor of the ELBS model and compared with the SIF model and HIH model.

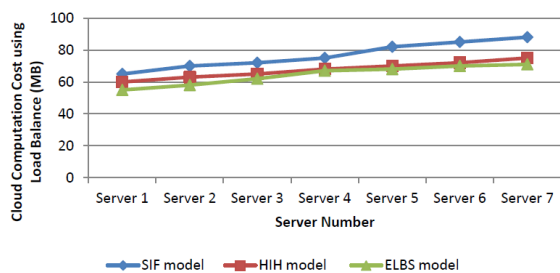


Fig. 7. Measure of cloud computation cost using load balance factor.

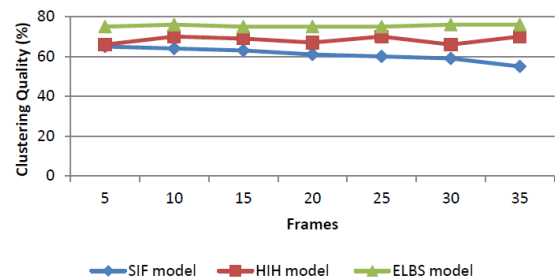


Fig. 8. Measure of clustering quality.

Fig. 7 illustrates the Cloud Computation Cost using Load Balance and compared with two other works SIF model, HIH model. The cloud computation cost using ELBS model is improved as it obtains the cluster head each time when the imbalances occur with respect to load threshold. The load balance factor is 18 – 28 % improved using ELBS model when compared to the SIF [9] model and 10 – 14 % improved than using the HIH [7] model.

Table 6. Tabulation of Clustering Quality

Frames	Clustering Quality (%)		
	SIF model	HIH model	ELBS model
5	65	66	75
10	64	70	76
15	63	69	75
20	61	67	75
25	60	70	75
30	59	66	76
35	55	70	76

Table 6 and Fig. 8 describe the clustering quality with respect to the number of frames in the cloud computing paradigm. The HIH model [7] did not integrate with the clustering model, so it achieved lesser efficiency when compared to the ELBS model. The clustered family of servers using ELBS model selected a computing server to satisfy the computational demands. As a result, the ELBS model improved the quality of cluster to 6 – 15 % when compared to SIF [9] model and HIH [7] model.

To conclude, as the balancing of load is considered as the prime factor, it has to be analyzed in order to maximize the time taken to respond for the service invocations and minimal bandwidth utilization on the basis of tasks being assigned. This is accomplished with the optimal management of the energy by minimizing the imbalances occurring in load in the cloud computing paradigm based on cluster. The experimental results show a significant improvement of time taken to respond, utilization of bandwidth and energy in an optimal manner for increasing number of users with higher level of scalability achieved according to the size of work.

5. Conclusion

The aspects of load balancing in VM cloud computing paradigm presented a load balancing approach to optimize utilization of energy and bandwidth. The ELBS model provides load balancing in VM data center. Moreover with the distribution of information between different servers congestion and deadlock is avoided and information losses due to the overflow in cloud infrastructure are mitigated. In addition, the ELBS model collects the information about the current load of other family using a progressive query messaging. As a result, the quality of service is improved in ELBS model by minimizing the delays with respect to communication and information losses due to congestion. The validation results, obtained from the VM cloud data center confirm the improvements in load balancing factor. The experimental result of ELBS model is compared with the existing SIF model and HIH model to achieve the minimal utilization of energy and bandwidth. The performance tradeoff with respect to the throughput is also improved in ELBS model with minimal time taken for responding to the requirements. Clustering quality is also improved to approximately 8.71 % using load balancing factor.

References

- [1] Qian, W., Cong, W., Kui, R., Wen, J. L., & Jin, L. (2011). Enabling public auditability and data dynamics for storage security in cloud computing. *Transactions on Parallel and Distributed Systems*.
- [2] Selvarani, S., & Sudha, S. G., *Improved cost based algorithm for task Scheduling in cloud computing*.
- [3] Imad, M. A., & Anbang, R. (2013). Towards trustworthy resource scheduling in clouds. *Transactions on Information Forensics and Security*.
- [4] Kazuki, M., & Shin, I. K. (2011). Evaluation of optimal resource allocation method for cloud computing environments with limited electric power capacity. *Proceedings of the International Conference on Network-Based Information Systems*.
- [5] Karthik, K., Jing, F., Yamini, N., & Yung, H. L. (2011). Resource allocation for real-time tasks using cloud computing.
- [6] Muhammad, A. A., Ryo, S., & Rajesh, G. (2012). Energy efficient geographical load balancing via dynamic Deferral of workload.
- [7] Ming, L., Shucheng, Y., Yao, Z., Kui, R., & Wen, J. L. (2012). Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. *Transactions on Parallel and Distributed Systems*.
- [8] Jian, F. Y., & Wen, S. L., (2009). Calibrating resource allocation for parallel. *Processing of the Analytic Tasks IEEE International Conference on e-Business Engineering*.

- [9] Avinash, M., Mukesh, M., Sanket, D., & Shrisha, R. (2011). Energy Conservation in cloud infrastructures. *IEEE Transaction on Cloud Computing*.



N. R. Ram Mohan received the B.E degree from Anna University, Chennai, Tamil Nadu, India, in 2007 and M.Tech degree from Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India, in 2009. He is currently pursuing the Ph.D degree in computer and information technology, Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India. His current interest span several areas grid computing, information security and cloud computing



E. Baburaj received the M.E. degree in computer science and engineering from Madurai Kamaraj University, Madurai, Tamil Nadu, India, in 2002 and the Ph.D degree in computer science and engineering from Anna University, Chennai, Tamil Nadu, India, in 2009. Currently he is a professor, Department of Computer Science and Engineering, Sun College of Engineering and Technology Nagercoil, Tamilnadu, India. His research interests include adhoc networks, network security and cloud computing. He has published over 20 papers in international journals.