SERVICE LEVEL AGREEMENT AWARE ENERGY OPTIMIZED SCHEDULING ALGORITHM FOR CLOUD COMPUTING ENVIRONMENT

Murgesh V. Jambigi(1), M.V. Vijay Kumar(2), D.V. Ashoka(3), Prabha R.(4)*

(1)Dept. of Computer Science and Engineering, GMIT Davangere; (2)Dept. of Information Science and Engineering, Dr. A.I. T.; (3)Dept. of Information Science and Engineering, JSS Academy of Technical Education; (4)Dept. of Computer Science and Engineering, Dr. A.I.T.
Bengaluru, India

* Corresponding Author: e-mail: murgeshcs062@gmail.com

Abstract: This paper presents a heterogenous cloud computing environment for provisioning real-time (dynamic workload) services in a cloud computing environment. Moreover, this work also presents SLA Aware Energy Optimized (SAEO) Scheduling Algorithm to execute dynamic workload applications like the data intensive and scientific applications. The main aim of the SAEO is to bring good tradeoffs in minimizing computation time and energy consumption by employing Dynamic Voltage Frequency Scaling (DVFS) effectively utilizing system resource of cloud. SAEO achieves much better performance than existing DVFS-based scheduling in terms of computation time and energy efficiency.

Key words: Cloud Computing, DVFS, Heterogenous Computing, Environment, SLA, Task Scheduling Technique.

1. INTRODUCTION

Nowadays, the internet is converted to service-oriented platform that gives service as the cloud computing. Scheduling of the task plays an important role to improve the performance of the cloud services. Task scheduling notices the tasks provided by the cloud users to the Cloud Service Provider (CSP) on accessible resources. There are multiple virtual machines for multiple user tasks. The resources are provided as the services in the forms of the virtual machine (VM) in cloud computing environment, to the clients across the global based demand. If the load is heavy and the demand for the resources are very high, the services, the resource is utilized in a higher extend. The load should be distributed across the servers of the cloud computing environment based on the amount of dynamically available resources; this will improve the performance of the applications running in the virtual machines. The application of loads on the servers plays an important role, if the applied load is less it results in the inefficient utilization of the server and if the load is too high then the performance of the server is degraded. In order to have an optimal performance, the amount of the resources utilized should be very right. However, exiting system resource management is not efficient as it increases latency in data access time; resulting in higher energy consumption for workload execution. Further extensive carried out show, existing resource management and DVFS-based scheduling model are predominantly designed considering homogeneous cloud environment; thus, are not efficient in meeting data intensive application workload performance with energy efficiency requirement.

For overcoming research issues here, the SLA aware and energy optimized workload scheduling is presented. The SEAO is designed employing DVFS for bringing tradeoffs among meeting workload performance requirement meeting energy efficiency constraint. The tradeoffs constraint introduced in this work are multi-objective parameter such as total computation, reconfiguration and interaction energy. No prior work as considered such consideration employing heterogeneous cloud environment. The SEAO utilizes the idle resources in the cloud to improve the efficiency, minimize the energy consumption by an efficient way.

The contribution of research work is as follows:

- This paper presented SLA aware task scheduling approach for heterogeneous cloud computing environment.
- The SAEO brings good tradeoffs between minimizing power consumption and meeting task SLA prerequisite.
- The SAEO model achieves better processing time and energy efficiency performance than state-of-art multi-objective-based task scheduling approach.

2. LITERATURE SURVEY

This section presents a survey of various existing workload scheduling on distributed and cloud computing environment in recent times. In [1] two different methods are proposed: first, power best fit algorithm; in this the machine is considered as the least power consumption machine, and the scheduling tasks are the tasks. The second one is balancing approach of the load, which is built on the base of power frequency ratio of every resource. The computing capacity of the server is indicated by the power frequency. In [2] proposed a model to achieve the energy efficient job scheduling by balancing load in cloud data center. They are based on the traffic requirement of the applications of the cloud. The communication delays and the congestion on the networks are reduced. The tradeoffs model of energy efficiency and network awareness is proposed in [3]. The throughput of the job is improved by satisfying the quality of service. The hotspots are avoided, and reduce the number of computing servers. The feedback channels are used in the main network switch, to obtain the network awareness. The memory overhead and the commutation delay are less in this method. In [4] presented an optimal scheduling technique that reduces the power consumption and also fulfills the task response time constraints. It can also be termed as a greedy methodology, because it selects the most efficient servers in a minimal number in the process of scheduling. In real time the tasks are heterogeneous in nature, because the response time of the tasks is different and the energy consumed by each task is different from one another. The assignment is said to be optimal, when the energy consumption is minimal and the time taken for the completion of the task is minimum on a particular machine. A well analyzed algorithm have been developed in [5], which is based on the min-min algorithm, this uses the technique of scheduling of the load on the service that prioritizes the user. There are two categories of users' ordinary user and subscribed/prime user. The balance of the load is based on the make span time and maximum loaded resource of respective computational platform. There is a good satisfaction to the user, it makes the use resource more efficiently with improved make span performance.

In [6] discussed various dynamic algorithms in regard of the resources allocated in accordance with its need and load distribution with respect to computational servers available. The results obtained with the usage of Xen cloud platform. In the algorithm it is noted that the response time has improved and the better service for the applications which are using the machines virtually with the factors of consideration such as migration and scaling. In [7] particularly showed that using virtualization in cloud environment will aid obtaining better load management. The major issues of the allocation of VM to a dynamic task that is dependent on the application of QOS demand, and the support of the energy and saving of the cost by the optimizing the count of the servers in the clients is discussed in [8]. Considering Nash Bargaining solution, as a base; in [8], proposes a lower cost and dynamic VM allocation model. Using varieties of simulation, the proposed method is shown to have reduced the overall budget in running the servers, guarantees the QoS demand, and increases the utilization of the server in various dimensions of the server resources. In [9] has focused about the data payload balancing in the environment of cloud computing with the dynamic workload access in the multiple nodes for the guaranteed service that not even a single node defined will exceed the predefined data performance or it will be left free without utilization. Here load balancing in the cloud environment under consideration for the various task/workload is considered. Discussed about the significance of genetic algorithm (GA) based load balancing in the cloud environment. Similarly, in [10] has proposed IMOPSO (Improved Multi-objective particle swarm optimization) model. IMOPSO is a dynamically varying model for the resource allocation with respect to the requirements of the user with their importance for the virtual machine usage being a crucial role in cloud computing with the agenda of the betterment in the resource utilization with the performance enhancement. There are numerous key challenges that are affecting the dynamic management of the workload in a cloud environment which is termed to be heterogeneous [11]. The total resource usage cost is reduced based on the service application across the data centers that are distributed to meet the service demand to minimize the resource usage cost. Then, they planned a heterogeneity-aware dynamic application provisioning technique to reduce the consumption of energy, this also satisfies the objectives of the performance. At the end, they analyzed scheduling problem and gives a new way of solution and a scheme that powers the heterogeneous run-time task usage characteristics [11].

Different meta-heuristic techniques, for instance, tabu search, chemical reaction optimization (CRO), genetic algorithm (GA), ant colony optimization (ACO) calculation, and toughening are comprehensively used in DAG-based information concentrated and logical work process planning [12]. This philosophy generally delivers predominant booking quality when contrasted and heuristic strategy. This is a direct result of helpless inquiry adequacy and regular procedure calculation [13]. The following issues have been identified in the existing work related to work load and dynamic resources allocation of process [14]. First, the overhead involved is high in terms of memory usage and CPU. Second, underutilization of physical machines. Third, very limited work is carried out to bring good trade-off between energy efficiency, latency and computation time. Lastly, the state-of-art task scheduling model is designed considering homogenous cloud computing environment.

3. SLA AWARE ENERGY OPTIMIZED TASK SCHEDULING APPROACH FOR HETEROGENOUS COMPUTING ENVIRONMENT

In this section we describe our proposal on SLA aware energy optimized (SAEO) task planning approach for heterogeneous processing environment for executing High performance computing (HPC) application in distributed computing environment utilizing DVFS method based particular frequencies and their time allotments for each VM. In particular, the proposed technique will assist to manage the task overload segment which can be represented as $\{P_k t_{sk},\ s = 1, \dots \dots \dots . . N\ , k = 0, \dots \dots \dots R\}$ and the terminal connection for the variable transmitting of the information given by $\{Q_s,\ s = 1, \dots \dots \dots . N\}$, which are the rates that are factors of consideration for reconfiguration, computation with the interaction energy expressed in terms of joule as

$$\gamma_l \triangleq \sum_{s=1}^{N} \gamma_{T_{\mathbb{C}}}(s) + \sum_{s=1}^{N} \gamma_{F_{\mathbb{C}}}(s) + \sum_{s=1}^{N} \gamma_{\mathbb{M}_{\mathbb{C}}}(s), \tag{1}$$

For DVFS method, the working recurrence for each VM lies in the little scope of unmistakable frequencies. The ideal working recurrence can be chosen by exchanging the CPU frequencies of VMs over a different scope of conceivable time-frames. Be that as it may, because of the presence of unmistakable frequencies a non-raised issue can happen which can be figured out as, where, $\gamma_{F_{\mathbb{C}}}(s)$ gives the cost of reconfiguration for VM(s) which will be utilized for the block-based processing and for the constraint $I_t$in SLA which provides interaction time. $\{\mathcal{A}(s),\ s = 1, \dots \dots \dots . N\}$ Which gives the representation of the interaction delay for a simple communication which relative to the consumption of energy for single interaction in a virtual connection? In regard of DVFS technology for the cloud-based computing, for each and every virtual machine operating frequency in the nearest range of unique frequencies. The operating frequency for the functioning of each node under consideration for the matching with the frequencies with the CPU for virtual machines in the nearest range for possible communication in the fixed time frame. But due to the presence of different frequencies a design challenge may be encountered for the sorting of usage for the switches of the virtual machine with the current unique frequency for the completion of the allotted load. Hence the required time is circulated equally into indefinite variables dependent on time represented as $R + 1$. Which will result in the unique frequencies for every virtual machine in the particular time slots and its frequency is represented as $\mathbb{f}_k$. The operating systems will make an account of the respective servers; hence it can allocate the task for the new job which will be newly arrived through the gateway. The data information is very important for the data handing over to the next node for processing and acts as one of the servers to process so that the cumulative consumption of energy will be minimized with the corresponding execution time and hence the above-mentioned scenarios can be termed as follows

$$\min_{\{Q_s, t_{sk}\}} \sum_{s=1}^{N} \sum_{k=1}^{R} (G_a \mathbb{C}_{\mathbb{e}} \mathbb{f}_k^3 t_{sk}) + \sum_{s=1}^{N} \gamma_{F_{\mathbb{C}}}(s) + \sum_{s=1}^{N} \sum_{k=1}^{R} 2\left(E_s^{\mathbb{M}_{\mathbb{c}}}(Q_s) P_k t_{sk} \,.\, (Q_s)^{-1}\right), \tag{2}$$

where, it is subjected to,

$$\sum_{s=1}^{N} \sum_{k=1}^{R} P_k t_{sk} = M_l, \tag{3}$$

$$\sum_{k=1}^{R} t_{sk} \leq I\,,\ S = 1, \dots \dots .. N, \tag{4}$$

$$\sum_{k=1}^{R} 2 P_k t_{sk} \,.\, (Q_s)^{-1} + I \ \leq I_t,\ s = 1, \dots \dots \dots . N, \tag{5}$$

$$\sum_{s=1}^{N} Q_s \leq Q_t. \tag{6}$$

In regard of the above equation which can be discussed as follows. The Eq. (2) depicts the representation of the interaction cost and energy consumption for the switching frequencies depending on the load for processing on the incoming data, it is also noted for consideration the Eq. (3) which is the formulation of the computing rates collectively of every virtual machine in regard of unique time slots depending on the M_l for incoming job allocation. But in regard of Eq. (4) & (5) equations which will interpret the factor I for the highest time for the data processing. The consumed energy in total and the time of interaction in dependency with the I_t as the constraint of SLA divided into Eq. (4) & (5), respectively and it will coordinately denote the computational and interaction cost as well. The Eq. (6) equation is the orientation of the total amount information communicated through the total information volume and the information center without the bypassing the absolute limit of the organization data focus, consequently this condition will also resemble the terminal connection with regard of the total load bandwidth with the fine tuning of the total operating virtual machine in accordance with the provided work load. To overcome the challenges of the non-convex, hence we make the separation into three remarkable occasions, for example, collaboration cost, recurrence reconfiguration and calculation cost, this load of occasions are booked independently for the achieving of the efficient execution for the attainment of the minimal energy. Hence with the above discussion the optimization issues in regard of computations are listed as follows

$$\min_{t_{sk}} G_a \mathbb{C}_{\mathbb{e}} \sum_{s=1}^{N} \sum_{k=0}^{R} \mathbb{f}_s^3 t_{sk}\,, \tag{7}$$

With the detailed discussion till now the few observations can be noted that the for the linear representation of the control parameter t_{sk} is done in the Eq. (7) and can be further deduced using the Eq. (3) and (4). In similar manner the Q_s and $P_k t_{sk}$ are the non-convex aware variables for the interaction with the optimizations can be summarized as follows

$$\min_{Q_s} 2\left(E_s^{\mathbb{M}_{\mathbb{c}}}(Q_s) P_{sk} t_{sk} \,.\, (Q_s)^{-1}\right) \tag{8}$$

The optimization problem of Q_s and $P_k t_{sk}$ is solved by using Eq. (5) and (6) or more optimal solution is obtained using Eq. (9) as follows

$$\sum_{s=1}^{N} \sum_{k=1}^{R} 2\left(E_s^{\mathbb{M}_{\mathbb{c}}}(Q_s) P_k t_{sk} \,.\, (Q_s)^{-1}\right) = (I_t - I) \sum_{s=1}^{N} \sum_{k=1}^{R} E_s^{\mathbb{M}_{\mathbb{c}}}.\ (2P_{sk} t_{sk} \,.\, (I_t - I)^{-1}). \tag{9}$$

The improvement issue in (1) can be figured out as,

$$M_l \leq Q_t \,.\, (I_t - I). (2)^{-1}, \tag{10}$$

$$M_l \leq \sum_{s=1}^{N} I P_R \ . \tag{11}$$

where, Eq. (10) and (11) are fundamental and reasonable for the achievability of streamlining issue happened in the Eq. (1). Presently, for reconfiguration cost $\gamma_{F_{\mathbb{C}}}(s)$ can be partitioned into two parts as external and internal reconfiguration cost. Initially $\mathbb{f}_k$ to $\mathbb{f}_{k+h}$ are the distinct frequency variation, in which the $\mathbb{f}_{k+h}$ factor h will be the movement for the outreach of the active distinct frequency for the entire $VM(s)$. Secondarily for the operation of switching among the active distinct frequency components for the respective time period in view of the VM(s). Hence the desired system will demand for the definition of the combination of switching cost internally and external as well among the virtual machines which are termed using following equations

$$h_a \sum_{s=1}^{N} \sum_{h=0}^{H} (\Delta \mathbb{f}_{sh}^2), \qquad h \in \{0,1, \dots \dots \dots .. H\}, \tag{12}$$

where, $H \leq R$represent overall active distinct frequencies for every $VM(s)$.

$$\sum_{s=1}^{N} \gamma_{F_{\mathbb{C}}} = h_a \sum_{s=1}^{N} \sum_{h=0}^{H} (\Delta \mathbb{f}_{sh})^2 + h_a \sum_{s=1}^{N} \mathbb{E}_{\mathbb{C}}. \tag{13}$$

where, $\mathbb{E}_{\mathbb{C}}$ addresses the underlying distinct frequency actively in the accompanying showing up task load, for the improvement of the energy in the expansion of execution at the pinnacle level. Thus, the trade-off among execution and energy utilization can be accomplished utilizing our proposed SLA aware energy improved planning approach which is tentatively demonstrated beneath.

4. RESULTS AND DISCUSSIONS

In this segment, for the assessment of execution and power utilization, different outcomes are shown utilizing our proposed SLA aware energy enhanced scheduling approach dependent on DVFS strategies. Here, we have assessed execution time thinking about various positions as 30, 50, 100, and 1000. Various graphs are plotted thinking about time, number of jobs, power utilization and so forth. Different boundaries are considered to assess execution time and resource utilization. The SAEO model is trained on Inspiral logical dataset. The SAEO model is executed utilizing Java programming language. The model is conveyed on 64-bit quad core processor with 16 GB RAM on Windows 10 working framework.

Performance Evaluation: The proper scheduling of the task in the cloud environment will result in the higher throughput for the better resource utilization which will avoid the tasks overload for the better interaction with the users. As depicted in the Figure 1, the performance result is obtained with the SAEO and existing DVFS method in regard of the complete execution with the consideration of varying VM size and different job size of Inspiral workflow. The job size of Inspiral workload is 20, 30, 100, & 1000. With the obtained results, it is observed with the total execution time of the SAEO technique with the workflow Inspiral 30, 50,100 and 1000 will be 1344.12 sec, 1419.89 sec, 2563.76 sec and 11859.21 sec respectively. In general, the outcome achieved it tends to be seen proposed SAEO reduce average total execution time by 92.64% when contrasted with standard model [17, 18].

With respect to the Figure 2, it is observed that the performance outcome achieved in the proposed SAEO in comparison with the DVFS considering the absolute execution time upon consideration with the various job size also refereed as task size for the workflow execution. With the same 4 job sizes the average execution time for the SAEO would be 44.804sec, 28.3978sec, 25.6376 and

11.85921sec for the 30, 50, 100 and 1000, respectively. From the general result accomplished it is very well seen that the proposed SAEO diminish normal execution time by 91.46% when contrasted and standard DVFS model [17, 18].

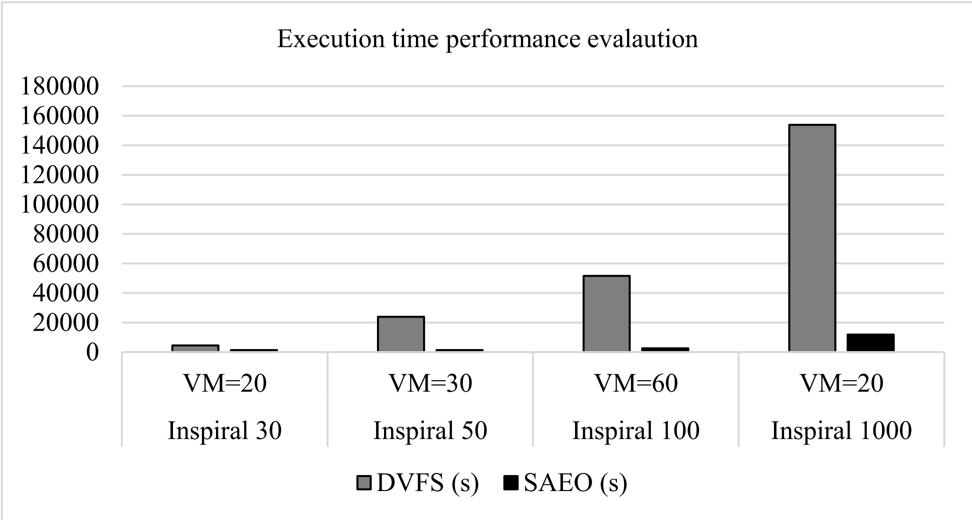

Figure 1. Execution Time Comparison of SAEO over existing DVFS strategy utilizing logical responsibility Inspiral.

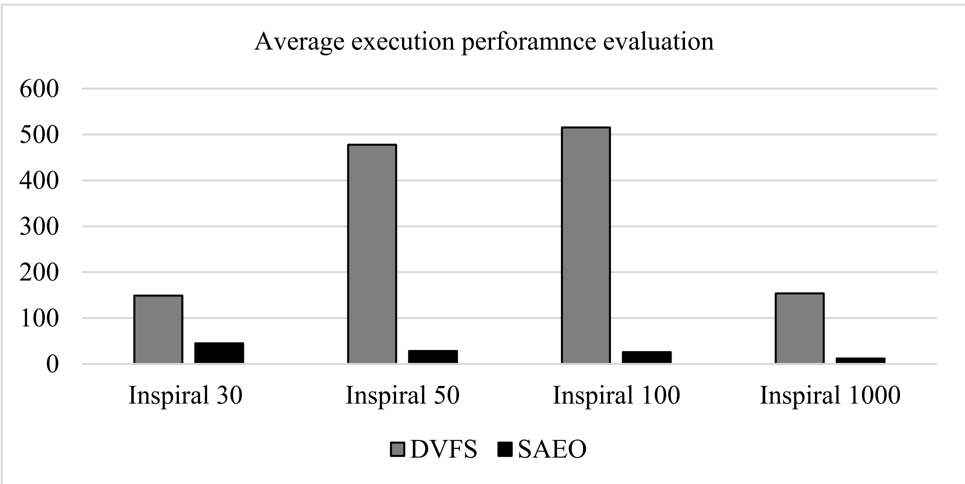

Figure 2. Normal Execution Time of SAEO over existing DVFS strategy utilizing logical responsibility Inspiral.

In accordance with the Figure 3, which discussed about the performance outcome for the proposed SAEO strategy when compared with the DVFS technique when focused in terms of the average power for the similar job patterns and obtained the power as follows 704.1820 watts, 829.3598 watts for the 30 and 50 job sizes, whereas 1493.822 watts, 13653.4518 watts for 100 and 1000, respectively. From overall result attained it can be seen proposed SAEO reduce average power consumption by 19.51% when compared with standard DVFS model.

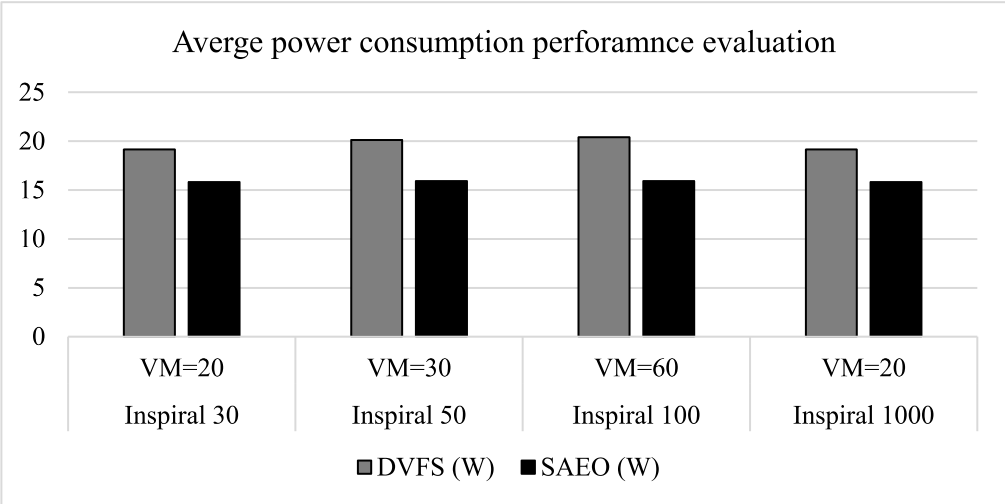

Figure 3. Power consumption of SAEO over existing DVFS strategy utilizing logical responsibility Inspiral.

In accordance with the Figure 3, which discussed about the performance outcome for the proposed SAEO strategy when compared with the DVFS technique when focused in terms of the average power for the similar job patterns and obtained the power as follows 704.1820 watts, 829.3598 watts for the 30 and 50 job sizes, whereas 1493.822 watts, 13653.4518 watts for 100 and 1000, respectively. From overall result attained it can be seen proposed SAEO reduce average power consumption by 19.51% when compared with standard DVFS model.

The Table 1, shows the SEAO novelty with respect to existing scheduling model such as DVFS [18] and Energy minimized workload scheduling (EMWS) algorithm [19]. From Table 1, we can see the DVFS supports workload execution for small to medium workload size; however, the EMWS and proposed SEAO supports execution of larger workload. Nonetheless, the EMWS is designed focusing minimizing energy; as result very poor makespan is result. On the other side the proposed SEAO model brings good balance between energy reduction and performance efficiency for small and large workload size and meets SLA (i.e., deadline) prerequisite of applications.

Table 1. Comparative analysis of SEAO over existing scheduling algorithm.

	DVFS [20], 2019	***EMWS [21], 2021***	***SAEO***
Cloud model adopted	*Heterogeneous computing environment*	*Heterogeneous computing environment*	*Heterogeneous computing environment*
Multi-objective function adopted	*Yes*	*No*	*Yes*
SLA considered	*No*	*No*	*Yes*
Performance metrics	*Cost and makespan*	*Energy*	*Energy and makespan*
Data-intensive and scientific workflow supported	*Yes*	*Yes*	*Yes*
Workload size supported	*Small to medium*	*Small to large*	*Small to large*

In this section we describe our proposal on SLA aware energy optimized (SAEO) task planning approach for heterogeneous processing environment for executing High performance computing

4. CONCLUSION

In this paper, different task scheduling calculations, computations, execution have been, investigated and simulated and accomplished using the CloudSim. In the above section of the results, it can be seen that the SAEO model has performed better in terms of execution time, power and also has reduced the SLA violation in a cloud environment. The existing system DVFS when compared with our model consumed less time for the execution of the workload when the virtual machines kept increasing. In this paper, we have discussed the importance of the SLA in the cloud environment. This model has reduced power, energy consumption, execution time and executes a given workload task in a given deadline with providing proper SLA.

Future work would consider utilize resource more efficiently by reutilizing cache resource. However, designing cache resource reutilization-based task scheduling is challenging which the future work would consider emphasize it.

REFERENCES

[1] Rambabu Medara, Ravi Shankar Singh, Amit. Energy-aware workflow task scheduling in clouds with virtual machine consolidation using discrete water wave optimization, *Simulation Modelling Practice and Theory*, Vol. 110, 2021, pp. 1-16, https://doi.org/10.1016/j.simpat.2021.102323

[2] Usman Sana, M., & Li, Z. Efficiency aware scheduling techniques in cloud computing: a descriptive literature review. *Peer Journal Computer science*. Vol. 7, No. 50, 2021, pp. 1-18, https://doi.org/10.7717/peerj-cs.509

[3] Malik N., Sardaraz M., Tahir M., Shah B., Ali G., Moreira, F. Energy-efficient load balancing algorithm for workflow scheduling in cloud data centers using queuing and thresholds. *Applied Science,* Vol. 11, No. 5849, 2021, pp. 1-17, https://doi.org/10.3390/app11135849

[4] Mubeen A., Ibrahim M., Bibi N., Baz M., Hamam H., Cheikhrouhou O. Alts: An adaptive load balanced task scheduling approach for cloud computing. *Processes*. Vol. 9, No. 1514, 2021, pp. 1-15, https://doi.org/10.3390/pr9091514

[5] Rajnikant PN, Patel N. Improving energy efficiency and minimizing service-level agreement violation in mobile cloud computing environment. *Springer, Data Science and Intelligent Applications*. Vol. 52, 2021, pp. 391-397, https://doi.org/10.1007/978-981-15-4474-3_43

[6] Daid R, Kumar Y, Hu YC, Chen WL. An effective scheduling in data centres for efficient CPU usage and service level agreement fulfilment using machine learning. *Connection Science*. Vol. 33, No. 4, 2021, pp. 954-974. https://doi.org/10.1080/09540091.2021.1926929

[7] Singh J. Energy consumption analysis and proposed power-aware scheduling algorithm in cloud computing. *Springer, Intelligent Computing and Applications*. 2021, pp. 193-201, https://doi.org/10.1007/978-981-15-5566-4_17

[8] Gong, W., Yan, J., Nan, X. et al. Joint optimization of routing and VM resource allocation for multimedia cloud. *Multimedia Systems.* Vol. 25, 2019, pp. 355-369, https://doi.org/10.1007/s00530-019-00611-1

[9] Kaur, Sukhpreet and Jyotsna Sengupta. Load balancing using improved genetic algorithm (IGA) in cloud computing. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*. Vol. 6, No. 8, 2017, pp. 1229-1233.

[10] K. Ray, S. Bose and N. Mukherjee. A load balancing approach to resource provisioning in cloud infrastructure with a grouping genetic algorithm. *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT).* 2018, pp. 1-6, doi: 10.1109/ICCTCT.2018.8550885.

[11] Li, S., Pan, X. Adaptive management and multi-objective optimization of virtual machine in cloud computing based on particle swarm optimization. *Journal of Wireless Communications Network*. 2020, pp. 102, https://doi.org/10.1186/s13638-020-01722-4

[12] Xu, B., Peng, Z., Xiao, F. *et al.* Dynamic deployment of virtual machines in cloud computing using multi-objective optimization. *Soft Computing,* Vol. 19, 2015, pp. 2265–2273, https://doi.org/10.1007/s00500-014-1406-6

[13] Z. Zhong, J. He, M. A. Rodriguez, S. Erfani, R. Kotagiri and R. Buyya. Heterogeneous task co-location in containerized cloud computing environments. *2020 IEEE 23rd International Symposium on Real-Time Distributed Computing (ISORC).* 2020, pp. 79-88, doi: 10.1109/ISORC49007.2020.00021.

[14] J. R. Doppa, R. G. Kim, M. Isakov, M. A. Kinsy, H. Kwon and T. Krishna. Adaptive manycore architectures for big data computing: Special session paper. *2017 Eleventh*

IEEE/ACM International Symposium on Networks-on-Chip (NOCS). 2017, pp. 1-8, https://doi.org/10.1145/3130218.3130236

[15] G. Xie, G. Zeng, R. Li and K. Li. Energy-aware processor merging algorithms for deadline constrained parallel applications in heterogeneous cloud computing. *IEEE Transactions on Sustainable Computing.* Vol. 2, No. 2, 2017, pp. 62-75, doi: 10.1109/TSUSC.2017.2705183

[16] Z. Li, J. Ge, H. Hu, W. Song, H. Hu and B. Luo. Cost and energy aware scheduling algorithm for scientific workflows with deadline constraint in clouds. *IEEE Transactions on Services Computing*. Vol. 11, No.4, 2018, pp. 713-726, doi: 10.1109/TSC.2015.2466545.

[17] Z. Zhu, G. Zhang, M. Li and X. Liu. Evolutionary multi-objective workflow scheduling in cloud. *IEEE Transactions on Parallel and Distributed Systems*. Vol. 27, No. 5, 2016, pp. 1344-1357, doi: 10.1109/TPDS.2015.2446459

[18] Junlong Zhou et al.. Cost and makespan-aware workflow scheduling in hybrid clouds. *Journals of Systems Architecture*. Vol. 100, 2019, pp. 1-12, https://doi.org/10.1016/j.sysarc.2019.08.004, 2019.

[19] B. Hu, Z. Cao and M. Zhou. Energy-minimized scheduling of real-time parallel workflows on heterogeneous distributed computing systems. *IEEE Transactions on Services Computing*. 2021, pp. 1-1, doi: 10.1109/TSC.2021.3054754.

Information about the authors:

Murgesh V Jambigi – Murgesh V Jambigi working as Assistant Professor in department of Computer Science and Engineering in GMIT Davangere. He is research scholar in V.T.U and his areas of interest are cloud Computing, Big Data and IOT. He is awarded as best keynote Speaker on Cloud Computing.

Dr. M. V. Vijaykumar – Dr. M.V. Vijayakumar is a reformer educationist, renowned researcher and administrator. Presently working as a Vice Principal, professor and HOD in ISE the Dr. Ambedkar Institute of Technology, Bangalore, India and in his research interest and Curiosity joined Full time PhD course in United Kingdom.

Dr. D. V. Ashoka – Dr. D. V. Ashoka working as a Dean (Research) JSSATEB & Professor in the department of Information Science and Engineering, JSS Academy of Technical Education, Bangalore. He worked as Professor and Head, Department of CSE/ISE in various reputed Engineering colleges of Karnataka, India.

Dr. Prabha R – Dr. Prabha R working as Professor in department of Computer Science and Engineering in the Dr. Ambedkar Institute of Technology, Bangalore and her research areas of interest are cloud Computing, Big Data and Machine Learning and she has published various research articles in reputed conference and journals.

Manuscript received on 22 December 2021