International Journal of Distributed and Parallel Systems (IJDPS) Vol.10, No.1, January 2019

REAL-TIME ADAPTIVE ENERGY-SCHEDULING ALGORITHM FOR VIRTUALIZED CLOUD COMPUTING

D B Srinivas[1], H K Krishnappa[2], Rajan M A[3], Sujay N. Hegde[4]

[1,4]Nitte Meenakshi Institute of Technology, Bangalore, India
[2]R V College of Engineering, Bangalore, India
[3]TCS Research and Innovation, Bangalore, India

ABSTRACT

Cloud computing becomes an ideal computing paradigm for scientific and commercial applications. The increased availability of the cloud models and allied developing models creates easier computing cloud environment. Energy consumption and effective energy management are the two important challenges in virtualized computing platforms. Energy consumption can be minimized by allocating computationally intensive tasks to a resource at a suitable frequency. An optimal Dynamic Voltage and Frequency Scaling (DVFS) based strategy of task allocation can minimize the overall consumption of energy and meet the required QoS. However, they do not control the internal and external switching to server frequencies, which causes the degradation of performance. In this paper, we propose the Real Time Adaptive Energy-Scheduling (RTAES) algorithm by manipulating the reconfiguring proficiency of Cloud Computing-Virtualized Data Centers (CCVDCs) for computationally intensive applications. The RTAES algorithm minimizes consumption of energy and time during computation, reconfiguration and communication. Our proposed model confirms the effectiveness of its implementation, scalability, power consumption and execution time with respect to other existing approaches.

KEYWORDS

Virtual Machine (VM); Virtualized Data Centers (VDCs); Quality of Service (QoS); Dynamic Voltage and Frequency Scaling (DVFS); Real Time Adaptive Energy-Scheduling (RTAES).

1. INTRODUCTION

Increased availability of the cloud models and allied developing models allows cloud service providers to provide easier and cost effective services to the cloud users. The cloud computing model generally depends upon virtualization technology, which allows multiple operating systems (OSs) to share a hardware system in an easy, secure and manageable way.

As more and more services are digitized and added to the cloud, cloud users have started using cloud services. Thus, the computing resources in cloud consume significant amount of power to serve user requests. This is evident from the following facts. Middle size data centers consume 80000 kW [1]. It is predicted that computing resources consumes nearly 0.5% of the world's total power usage [2], and if current demand continues, this is projected to quadruple by 2020. From these observations, it is evident that optimizing energy consumption is an important issue for cloud services in a data center. Energy minimization can be done in two ways (i) Designing energy-efficient processors. (ii) Minimizing the energy consumption by task scheduling.

To reduce energy consumption in the cloud environment, a Dynamic Voltage and Frequency Scaling (DVFS) approach is used. A DVFS is the hardware model, that has been used to increase

DOI:10.5121/ijdps.2019.10101

and decrease the frequency of CPUs depending upon the user's requirements [3] and is widely used in modern processors. The optimistic correlation of energy consumption and the CPU frequency can control the energy consumption. The use of a DVFS in a processor can dynamically change the working frequency and leads to a diverse consumption of energy. A lower frequency that consumes less energy is always not correct because a low frequency that maximizes the execution time of a task; also depends upon both the execution time and the power. In [4, 5], the authors specify that there are some optimum frequencies for a given processor, where the consumption of energy can reduced during the execution of a task.

In a data center, operating a processor at an optimal frequency results in minimum energy consumption. Generally user application contains set of tasks. If all tasks have been executed under the optimum frequency then there is a chance that some tasks may not meet their deadline. Therefore, efficient task allocation with a suitable frequency is a major challenge, since the optimum suitable frequency for each processor can optimize the total consumption of energy. An optimal DVFS-based strategy for task allocation should minimize the overall consumption of energy and meeting the required QoS (i.e., perform the task on time). A cost-constrained Heterogeneous Earliest Finish-Time (HEFT) model was studied in [6, 7], and these algorithms provides the optimum budget in different scenarios. The execution cost minimization for scientific workflows with a deadline constraint is considered and the meta-heuristic algorithm is proposed in [8]. The investigation of servers' power consumption with different numbers of VMs and various utilization has discussed in [9]. A previous study of the 'worst case of execution cycle/time' (WCEC/WCET) was presented in [10-12], where the execution time of the upper bound at the maximal frequency acts as the task workload. However, the WCET generally does not contest the running workload of a real task, which may result in a difference between the real energy savings and the theoretical results. Similarly in [13, 14] have made use of the probability function based on WCET in a workload to enhance the effectiveness of scheduling tasks.

Assessing the energy utilized by Virtualized Data Centers (VDCs) is one emerging technique that aims to implement an effective energy management for a virtualized computing platform [15, 16]. The main aim is to deliver high-class service to large number of clients, while minimizing the overall networking and computing energy consumption. In paper [17], they considered the power management and optimal resource allocation in a VDC switch with heterogeneous applications; but they did not control for the internal and external switching to server frequencies, which causes degrades the performance. In this paper, we propose the RTAES algorithm by manipulating the reconfiguration proficiency of Cloud Computing-Virtualized Data Centers (CCVDCs) that process massive amounts of information/data in parallel processing. RTAES algorithm minimizes energy consumption in the computational, reconfiguration and communication costs; under the scenario of parallel processing data in a cloud computing environment, while satisfying the service level, which is expressed as the maximized time to process the job (including communication and computational times). The main contribution of this research is; we consider the energy objective approach, which is a non convex function. In addition, we also propose a mathematical function to convert the non-convexity into a convex function. Moreover, our proposed model is easy to implement, scalable and independent of workload scheduling.

The organization of the paper is as follows. Section 2 presents the existing state of art methods related to our research in the cloud environment. Section 3 describes proposed RTAES methodology. In Section 4 we discuss the experimental results and the performance analysis of the proposed methodology. Finally the paper is concluded in Section 5.

2. RELATED WORK

The present cloud computing environment provides the necessary QoS to attract users from different segments. In the cloud computing, several resources such as; storage, memory, CPUs etc, where key resources are provisioned and then, they are leased to the users/clients for a guaranteed QoS. In [18], authors have presented a novel-SLA (Service Level Agreement) framework for cloud computing. The framework contains the reinforcement learning (RL) to develop the hiring policy of VM, which can be considered to assure the QoS. In [19], authors have proposed a DVFS approach that counters the trade-off between different energy consumption level and the performance dilapidation. CPU re-allocation approach is presented in [20] for effective energy-management in real-time services. Energy aware resource provisioning was considered in [21]. The proposed framework considers both server and communication network's energy consumptions in order to provide the complete solution. Efficient energy resources allocation model for real time services in the cloud environment is presented in[22]. There are several VMs and hosts that have been applied in order to minimize energy consumption. The scheduling of VMs on servers to meet the energy saving parameter in the cloud computing environment was discussed [23, 30], where a VM-scheduling approach was proposed in order to schedule the VMs in a cloud cluster environment. Because of the higher time complexity, this model is not as effective. Our previous work in [27], we measure the energy consumption of a parallel application on computational grids based on processor clock cycles to understand energy consumption issues.

Therefore, a novel scheduling approach called EVMP (Energy-aware VM-Placement) was considered to schedule the VMs, which was more effective at reducing the power consumption.

In [8, 24], authors have proposed a meta–heuristic algorithm to minimize cost with deadline constraints. The investigation of server power consumption with different number of VMs and various utilizations was discussed in [9]. A VM-scheduling algorithm, which ensures the efficient energy budget of an individual VM, was proposed in [25]. A cost constrained model through Heterogeneous Earliest Finish-Time (HEFT) is presented in [23], in which the algorithm provides the optimum budget in various scenarios. A similar approach is proposed in [6], where authors proposed the heterogeneous Budget Constrained Scheduling (BCS) approach that provided the efficient cost model by adjusting the costs and budget ratio within the assigned tasks workflow. One multi-objective HEFT approach is a Pareto heuristic-based algorithm [7], which is an extension to HEFT that, upgrades the costs and time factors simultaneously in order to provide a scheduling trade-off between the objectives. The real-time scheduling of a VM in the cloud to reduce consumption of energy was briefly addressed in [24]. Moreover, the reduction of energy consumption can also minimize the carbon dioxide emissions (from the datacenter cooling system) [25].

Therefore, this study mainly focuses on the VMs scheduling in order to minimize the energy consumption in cloud environment.

3. PROPOSED RTAES ARCHITECTURE MODELING

There are several virtualized processing units that are interconnected through a single virtual-hop network is considered in the CCVDCs model. Here the central controller is used to manage all the devices.

An individual processing component executes the primary assigned task through its computing resources and local virtual-storage. Whenever a novel task request is submitted at the CCVDC, the performances of the location resource availability and admission control are performed by the virtual central controller. Figure 1 shows the considered RTAES architecture, which shows the computing and communication system model. The multiple numbers of VMs are configured in the CCVDC where they are interconnected by a limited-switched V-LAN (virtual local area network). Here, a star-shaped V-LAN is considered that guarantee the inter communication of VMs and that the switching network in Fig. 1 will work as a central node. The VM-manager generally operates both the V-LAN and VMs jointly and performs the task scheduling activity by dynamically allocating the use of communication resources and available virtual computing to virtual links and VMs.

Here, we consider that VMs that are deployed over the server can alter their service resources according to the prototype presented in [31-33], this prototype has been widely adopted in the 'private-cloud environment'. The high computation demands in the model tend to consider some large VMs in the place of several small VMs. In this paper, we consider a single VM that is positioned as on individual server to simplify the model, and for that server we can use all resources. In the physical server, the CPU functions at a frequency (a_i) at any considered time from the predefined existing set of frequencies that are present in the DVFS approach.

The computation of energy depends upon the VM states and the curve of the CPU energy. Here, the DVFS is used to hosting the physical servers in order to stretch the tasks processing times and reduce the consumption of energy by minimizing the CPU frequencies of the working VMs.

The purpose of an individual server is to work at various voltage levels and, functions at various CPU frequencies. The set of permitted frequencies is;

$$a(i) \in \{a_j(i)\}, with i \in \{1, \ldots, Z\}, j \in \{0,1, \ldots, B\} \tag{1}$$

Where, B is the number of the 'CPU-frequency' that is allowed at each of the VMs, $a_j(i)$ is the jth discrete frequency of VMi, and $j = 0$represents its idle state.

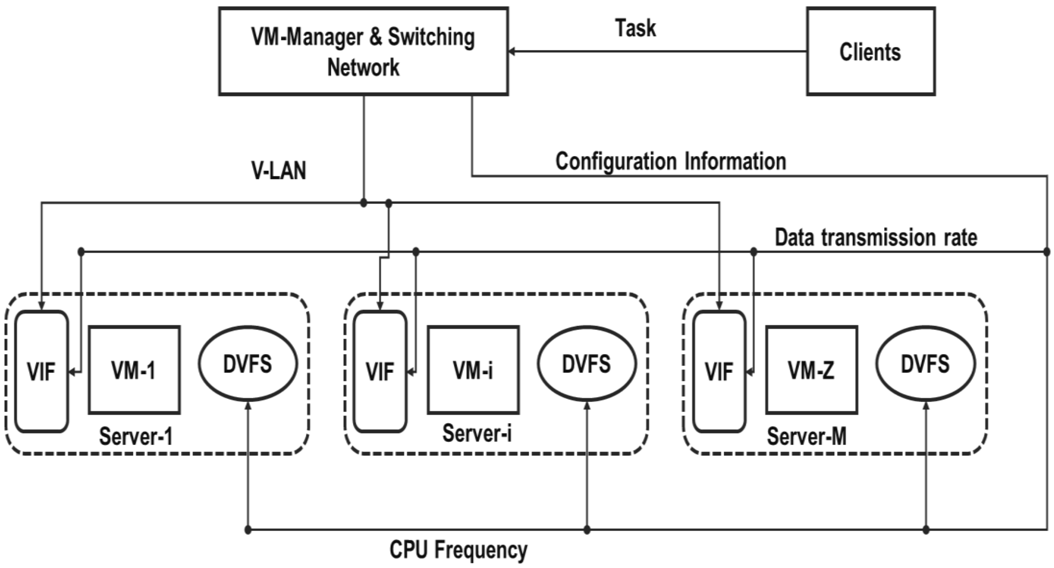

Figure 1. Proposed RTAES architecture.

The dynamic consumption of power C of hosting the CPU will increase with the 3rd power of the 'CPU frequency'. Therefore, in generic VM (i) the energy consumption is

$$\partial_{FC}(i) \triangleq \sum_{j=0}^{B} DE_{ce} a_j(i)^3 p_j(i)\,, [J] with \forall i \in \{1, \ldots, Z\} \tag{2}$$

where, $p_j(i)$ is the time with the jth VM operator at frequency$a_j(i)$.

The two costs are considered the switching frequency costs. In the first, the changing costs are from the internal switching at the discrete frequencies of VM (i) from$a_j(i)$ to $a_{j+f}(i)$.In the second, the costs are from the external switching through the final activated 'discrete frequency' for the upcoming job with a job-size ofG_{sz}. The list of discrete active frequencies for individual VMs in each upcoming task is input into the system, and the switching activity from the discrete active frequency to a different one disturbs the switching costs. The first discrete active frequency is $a_f(i)$, and the second one is $a_{f+1}(i)$. The difference between the frequencies is

$$\triangle\, a_f(i) \triangleq a_{f+1}(i) - a_f(i) \tag{3}$$

The switching costs are$H_c \triangle a_f(i)^2$, where the switching costs are calculate by a unit frequency switching as H_c (J/Hz^2). In the homogeneous VMs, the internal switching costs of all VMs are $k_e \sum_{i=1}^{Z} \sum_{f=0}^{F} [\triangle\, a_f(i)]^2$, where $f \in \{0, 1, \ldots, F\}$. $F \leq B$ is the number of discrete active frequencies at VM (i). Therefore, the total switching frequency is

$$\sum_{i=1}^{Z} \partial_{Sw}(i) \triangleq H_c \sum_{i=1}^{Z} \sum_{f=0}^{F} [\triangle\, a_f(i)]^2 + H_c \sum_{i=1}^{Z} E_c \tag{4}$$

The ‘external-switching costs’ (E_c) are considered by the multiplication of H_c with the last and first discrete frequencies’ quadratic differences.

The virtual link is responsible for the communication between VM (i) and the scheduler, which operates at $K(i)$ [bits/s] transmission rate (i.e., $i = 1, \ldots, Z$) and is equipped with a virtual network interface (VIF), as given in Fig. 1. The power consumed in the transmission is one-way, and the switching function at the $i-th$ virtual link is

$$L^{con}(i) = L^{T}_{con}(i) + L^{R}_{con}(i), [W] \tag{5}$$

where the power consumed through the transmitted VIF is $L^{T}_{con}(i)$ and the receiving power with the switching function related to VIF is $L^{R}_{con}(i)$. Here, we consider that the receipt and transmission are same, and it can be calculated as

$$L^{con}(i) = M_i\left(2^{R(i)/N_i} - 1\right) + L^{idle}(i), [W] \tag{5}$$

where

$$M_i \triangleq Q_0(i)N_i / r_i, [W/Hz] \tag{6}$$

r_i and N_i[Hz] are the transmission power bandwidth and the noise ‘spectral-power-density’ with the positive gain of the i^{th} link. Therefore, the one-way corresponding delay in transmission is

$$S(i) = \sum_{j=1}^{B} \frac{A_j(i)p_j(i)}{K(i)} \tag{7}$$

In addition, the corresponding energy at one-way communication is

$$\partial^{com}(i) \triangleq S(i)L^{com}(i)\ [Joule] \tag{8}$$

Here, our main aim is to minimize the computational energy and overall resulting communication. This can be shown as

$$\partial_{tot} \triangleq \sum_{j=1}^{Z} \partial_{FC}(i) + \sum_{i=1}^{Z} \partial_{Sw}(i) + \sum_{j=1}^{Z} \partial^{com}(i)\ \ [joule] \tag{9}$$

where ∂_{tot} is the overall computational energy that is formed by the different cost function of VM (i). Moreover, the problem is with the difficulty constraint $(\overline{E_t})$ allowed in the per-job ‘execution time’.

3.1. Proposed RTAES technique for Optimization Problem

The goal of our proposed work is to minimize the total consumption of energy (from the incoming workload) by choosing the optimal computational resources in order to complete the task’s execution, which depends upon the optimal bandwidth (for the consumption of energy) and the current load level while adopting the content dependent switching frequencies for individual VMs. The computation of resources consists of several (PMs) physical machines, where each one comprises a single or multiple cores, a local I/O network interface and memory. The resulting overall communications and energy include the $\partial_{FC}(i)$,$\partial_{Sw}(i)$ and $\partial^{com}(i)$ for Z number of VMs in the ∂_{tot}, which are estimated for individual VMs.

To optimize the problem, it is important to apply an additional parameter E_t, which is the threshold value in the task's processing operation. Therefore, the service level is split into two constraints, which are that the network-dependent time is less than 'or' equal to $\acute{E}_t - E_t$ and that the computational time is less than or equal to E_t. Therefore, the resulting equation of the optimization problem can be expressed as;

$$Minimize \sum_{j=1}^{Z} \partial_{FC}(i) + \sum_{i=1}^{Z} \partial_{Sw}(i) + \partial^{com}(i) \tag{10}$$

Equation (10) consists of three major terms to find the computational energy. The decision variable for optimizing the difficulty is $p_j(i)$, where the time duration is $p_j(i)$ for each VM operating at frequency $a_j$frequency and the transmission rate for each VM (i) is $K(i)$.

$$G_{sz} = \sum_{j=1}^{Z} \sum_{j=0}^{B} A_j(i) p_j(i) \tag{11}$$

The above equation (11) is a global constraint that insures that the overall task is split into Znumber of parallel tasks. The G_{sz} is incoming job size, which needs to spread over Z number of VMs during computation.

$$\sum_{j=1}^{Z} K(i) \le K_t \tag{12}$$

(12)

The inequality bandwidth in the above equation (12) ensures that the summation of the bandwidth at each VM should be less than the maximum global network's bandwidth.

$$\sum_{j=0}^{B} p_j(i) \le E_t, \qquad i = 1, \dots, Z \tag{13}$$

This is the computational time constraint T.

$$\sum_{j=0}^{B} \frac{2A_j(i) p_j(i)}{K(i)} \le \overline{E_t} - E_t, \qquad i = 1, \dots, Z \tag{14}$$

Equation (14) describes the constraint at the data exchange period/time.

$$0 \le p_j(i) \le E_t, i = 1, \dots, Z, j \in \{0, \dots, B\} \tag{15}$$

$$0 \le K(i) \le K_t, \forall i \in \{1, \dots, Z\} \tag{16}$$

Equation (15) guarantees the time of individual computing intervals (i.e., positive and less thanE_t). The final constraint ensures that the K control parameter (i.e., the channel communication rate) should be positive and less than the maximum network capacity. The power-cost at the end-to-end link comes from equation (8) and the outcome can be given as

$$\sum_{j=0}^{B} \frac{2L^{com}(i)A_j(i)p_j(i)}{K(i)} \tag{17}$$

Where $K(i) > 0$ and$\forall i \in \{1,\dots,Z\}$. withThe multi-variable coefficient value and arrow sign implies that

$$\sum_{j=0}^{B} \frac{2\,A_j(i)p_j(i)}{K(i)} \leq \overline{E_t} - E_t \rightarrow \left(\sum_{j=0}^{B} \frac{A_j(i)p_j(i)}{K(i)}\right) \leq \frac{\overline{E_t} - E_t}{2} \tag{18}$$

The above equation (18) is obtained from equation (13). Furthermore, to make it easier to optimize the problem, here we modify the 'second control' variable through $K(i)$with the other control variable$p_j(i)$as

$$\sum_{j=0}^{B} \frac{2\,A_j(i)p_j(i)}{K(i)} \leq \overline{E_t} - E_t \rightarrow K(i) \geq \sum_{j=0}^{B} \frac{2\,A_j(i)p_j(i)}{\overline{E_t} - E_t} \tag{19}$$

The outcome of equations (18) and (19) is applied to get the $\partial^{com}(i)$ end-to-end function that is dependent upon the control variable $R\left(U(i); p_j(i)\right)$, which can be modified by altering the control variable $U(i)$to the other 'control variable'$p_j(i)$ in eq. (20).

$$\partial^{com}(i) = R\left(U(i); p_j(i)\right) \triangleq V\left(a_j(i)\right) \tag{20}$$

Eq. (20) represents the formula for the adaptive energy communication in the end-to-end link, which depends upon the time variables' simulation for each VM and the main function $V(.)$.that satisfies the third term in eq. (12) is convex.

4. Result And Analysis

Over the past several years, the embedded processor demand has become extremely high worldwide due to the vast usage of network equipment, information devices, portable gadgets and digital instruments. Supreme performance is always required from the embedded equipment because it is used in our daily lives, such as in the multimedia 'digital-signal-processing' technique. However, these devices that are equipped with embedded processors have some drawbacks that affect their device's efficiency due to the consumption of high power by the devices. Moreover, there are performance imbalances between devices. Therefore, here we evaluate the performance of our proposed model and existing models with respect to power consumption. Several results are presented using our proposed RTAES algorithm that is based upon the DVFS approach. In this section, we have considered the different jobs of '30', '50','100' and '1000' in order evaluate the execution time and the '$Montage$' scientific dataset

(MSD) has been adapted to validate our model. The power consumption and execution time can be evaluated using several parameters, which are shown in below Table 1. This model is simulated using cloudsim and implemented on the Windows 10, 64-bit operating system and uses a computer with an i5 processor with 8 GB RAM and the programming language used for code is JAVA.

4.1. Comparative Analysis

In the present technical era, cloud computing has conquered the world market in several areas, such as trading, healthcare, the software sector and medical. Thus, the upcoming technologies are largely based on cloud computing processor operations because of its extensive demand. The lack of accurate resource utilization can affect their device efficiency due to their high power consumption; therefore, to address these issues, here we introduce the RTAES algorithm that is based upon the DVFS approach. The effective scheduling of tasks can improve users' interactions, avoid task overloads, and enhance resource utilization and throughput. Therefore, here, we presented the comparison of results with existing techniques with respect to the consumption of power and operational execution time.

Table 1: Various parameters comparison for proposed technique vs DVFS using MSDX

Parameters			**Total execution time (sec)**	**Energy consumption (Wh)**	**Power Sum (W)**	**Power Average (W)**
DVFS	**MSD-25**	**VM-100**	174.67	482.29	1663184	95.22
	MSD-50	**VM-80**	387.53	944.14	2953095	76.2
	MSD-100	**VM-60**	817.35	1839.12	4673928	57.18
	MSD-1000	**VM-40**	8591.39	70377.1	3.3E+07	38.16
RTAES	**MSD-25**	**VM-100**	57.72	120	417907	72.4
	MSD-50	**VM-80**	71.77	124.88	418104	58.25
	MSD-100	**VM-60**	101.91	143.79	449547	44.11
	MSD-1000	**VM-40**	777.04	1046.14	2328420	29.96

The several parameters that are used in Table 2 are the total simulation time, the energy consumption, the power sum and the power average. The scientific model of Montage was considered with different job sizes, such as MSD-25, MSD-50, MSD-100 and MSD-1000. Moreover, each MSD that we have considered was tested with different virtual machines (VMs) that are shown in Table1 above. The energy consumption values are 482.29Wh for MSD-25, 944.14Wh for MSD-50, 1839.12Wh for MSD-100 and 70377.1Wh for MSD-1000. As shown in Table 1, these values are substantially lower than those of the existing technique. Here, we obtained 75%, 86%, 92% and 98% less energy consumption with respect to the DVFS approach.

The total simulation times in the existing approach for different MSDs are 174.67, 387.53, 817.35 and 8591.39 sec, which consumed 66%, 81%, 87% and 90% more time compared to that of our proposed model respectively. Table 2 represents the Average Execution time (sec) comparison of the proposed technique with other existing techniques using MSD. The proposed RTAES approach calculates the average execution times for MSD-25 as 2.3 sec, MSD-50 as 1.4 sec, MSD-100 as 1.02 sec and MSD-1000 is 0.77 sec, which are substantially lower than those with existing techniques such as EMOA [26] and DVFS.

Table 1: Average execution time (sec) comparison of the proposed technique with other existing techniques using MSD

Datasets	**Number of nodes**	**Average Execution time (s)**		
		EMOA [34]	**DVFS**	**RTAES**
MSD-25	30	8.44	6.9868	2.3088
MSD-50	50	9.78	7.7506	1.4354
MSD-100	100	10.58	8.1735	1.0191
MSD-1000	1000	11.36	8.59139	0.77704

4.2. Graphical Representation

Here, we represent the graphical form of our obtained results. Figure 2 shows the total simulation time comparison of the existing DVFS model and the proposed RTAES approach using the Montage scientific model datasets for different numbers of nodes and VM numbers. Figure 3 shows the energy consumption comparison of the existing DVFS model and the proposed RTAES approach using the Montage scientific model datasets for different numbers of nodes and VM numbers. Figure 4 shows the power sum comparison of the existing DVFS model and the proposed RTAES approach using the Montage scientific model datasets for different numbers of nodes and VM numbers. Figure 5 shows the power average comparison of the existing DVFS model and the proposed RTAES approach using the Montage scientific model datasets for different numbers of nodes and VM numbers. Figure 6 shows the average execution time (s) comparison of the existing DVFS model and the proposed RTAES approach using the Montage scientific model datasets for different numbers of nodes and VM numbers.

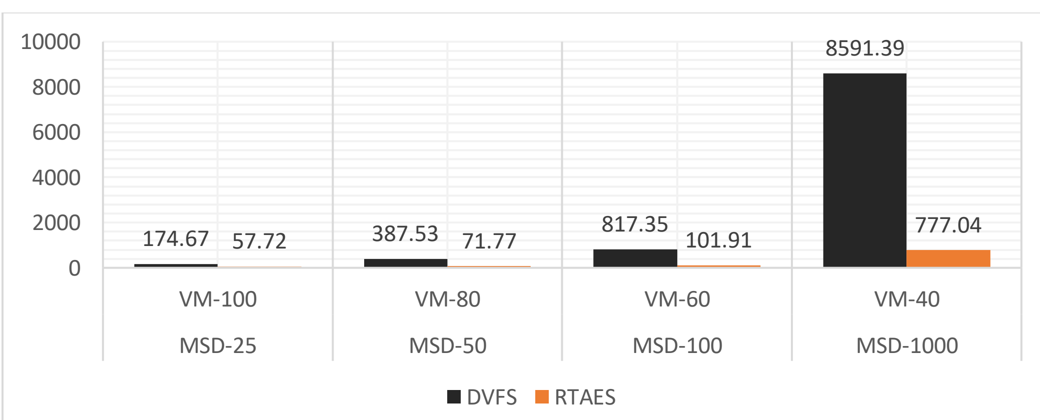

Figure 2: Total simulation time comparison of the existing and the proposed model using MSD

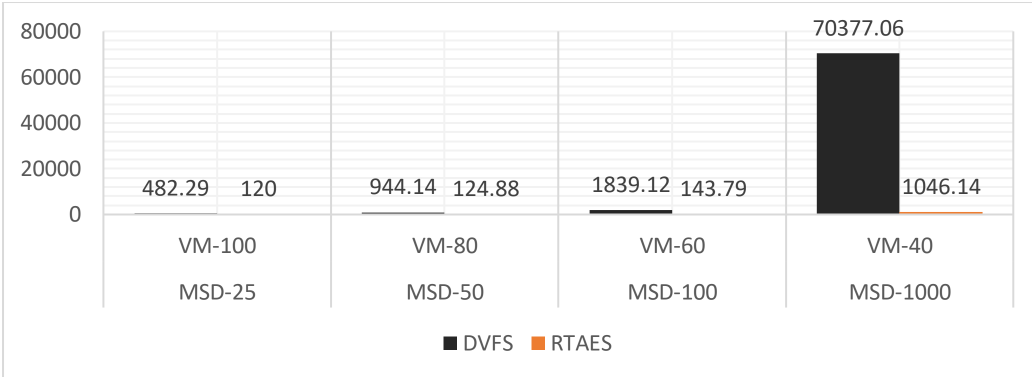

Figure 3: Energy consumption comparison of the existing and the proposed model using MSD

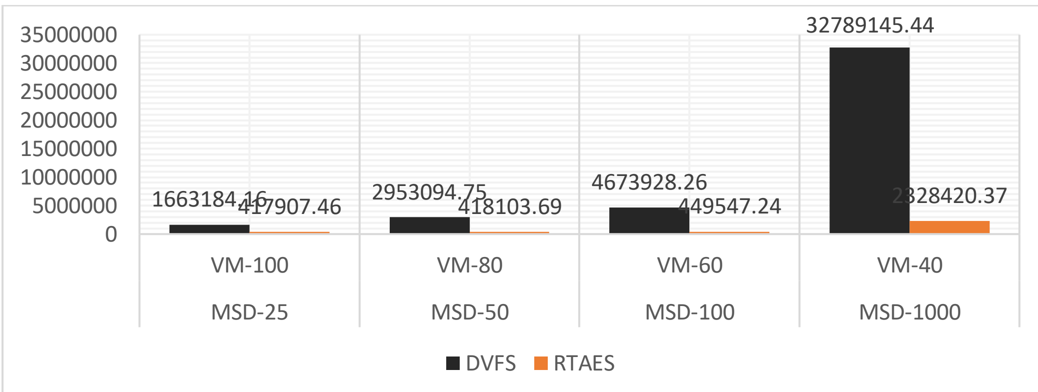

Figure 4: Power comparison of the existing and the proposed model using MSD

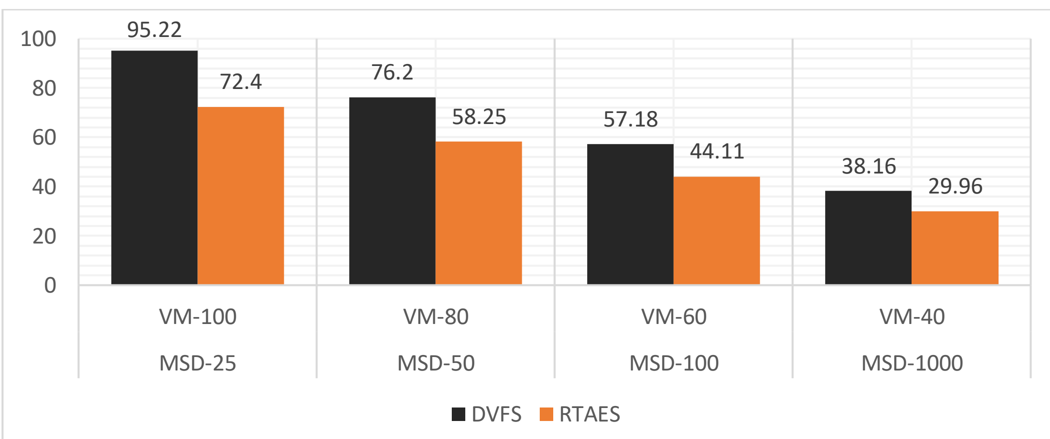

Figure 5: Power average comparison of the existing and the proposed model using MSD

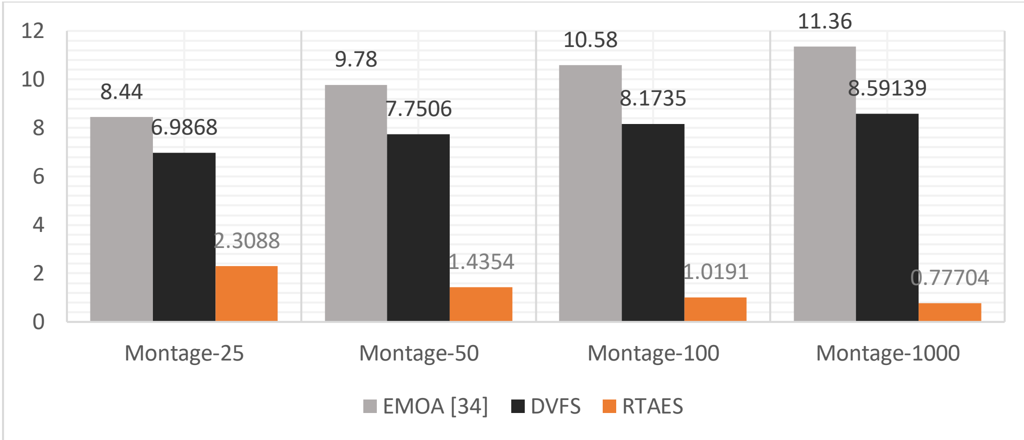

Figure 6: Average execution time (sec) comparison of the existing techniques and the proposed model using MSD

5. CONCLUSION

Task scheduling in embedded processors to minimize the consumption of energy in individual VMs, has essential significance. Therefore, in this paper, we proposed the real time adaptive energy-scheduling (RTAES) algorithm by manipulating the reconfiguration proficiency of cloud computing-virtualized data centers (CCVDCs) by processing massive amounts of information / data of an application. The RTAES algorithm minimizes the consumption of energy in the computational, reconfiguration and communication costs under the scenario of parallel processing data in a cloud computing environment. The RTAES algorithm is based on DVFS. We also tried to minimize three cost functions, including communication costs, computational costs, and switching costs, to improve the system's performance. The modeling details were presented above. The Montage scientific dataset (MSD) was used to validate our model with respect to other existing techniques. The above results were demonstrated in terms of the total execution time, the decrease in power consumption and the average power that was essential for processor operation. Using our proposed RTAES model, the total simulation times are 57.72 sec for MSD-25, 71.77 sec for MSD-50, 101.91 sec for MSD-100 and 777.04 sec for MSD-1000, values that are substantially lower than those of the existing DVFS technique. Similarly, the energy consumption values using our proposed RTAES model are 120Wh for MSD-25, 124.88Wh for MSD-50, 143.79Wh for MSD-100 and 1046.14Wh for MSD-1000, values that are substantially lower than those of the existing DVFS technique. Therefore, the results of our proposed RTAES model demonstrate its effectiveness in power consumption, performance and execution time with respect to other existing approaches.

REFERENCES

[1] Lizhe Wang, Gregor von Laszewski, Jai Dayal, Thomas R. Furlani, Thermal aware workload scheduling with backfilling for green data centers, in: IPCCC, 2009, pp. 289–296.

[2] William Forrest, How to cut data centre carbon emissions? Website, December 2008.

[3] Yongpan Liu et al. “Thermal vs energy optimization for dvfs-enabled processors in embedded systems”. In: Quality Electronic Design, 2007. ISQED’07. 8th International Symposium on. IEEE. 2007, pp. 204–209.

[4] J.-J. Chen and C.-F. Kuo, ``Energy-efficient scheduling for real-time systems on dynamic voltage scaling (DVS) platforms," in Proc. IEEE RTCSA, Aug. 2007, pp. 28-38.

[5] V. Devadas and H. Aydin, ``Coordinated power management of periodic real-time tasks on chip multiprocessors," in Proc. Green Comput. Conf., 2010, pp. 61-72.

[6] Arabnejad, H., Barbosa, J.G.: A budget constrained scheduling algorithm for workflow applications. J. Grid Comput. 12(4), 15 (2014)

[7] Durillo, J.J., Prodan, R.: Multi-objective workflow scheduling in amazon ec2. Cluster Comput. 17(2), 169–189 (2014)

[8] Q. Wu, F. Ishikawa, Q. Zhu, Y. Xia and J. Wen, "Deadline-Constrained Cost Optimization Approaches for Workflow Scheduling in Clouds," in IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 12, pp. 3401-3412, Dec. 2017.

[9] S. Chinprasertsuk and S. Gertphol, "Power model for virtual machine in cloud computing," 2014 11th International Joint Conference on Computer Science and Software Engineering (JCSSE), Chon Buri, 2014, pp. 140-145.

[10] T. AlEnawy et al., ``Energy-aware task allocation for rate monotonic scheduling," in Proc. IEEE RTAS, Apr. 2005, pp. 213-223.

[11] C.-Y. Yang, J.-J. Chen, T.-W. Kuo, and L. Thiele, ``An approximation scheme for energy- efficient scheduling of real-time tasks in heterogeneous multiprocessor systems," in Proc. DATE, 2009, pp. 694-699.

[12] E. Seo, J. Jeong, S. Park, and J. Lee, ``Energy efcient scheduling of realtime tasks on multicore processors," IEEE Trans. Parallel Distrib. Syst., vol. 19, no. 11, pp. 1540-1552, Nov. 2008.

[13] C. Xian and Y.-H. Lu, ``Dynamic voltage scaling for multitasking real-time systems with uncertain execution time," in Proc. ACM GLSVLSI, 2006, pp. 392-397.

[14] C. Xian, Y.-H. Lu, and Z. Li, ``Energy-aware scheduling for real-time multiprocessor systems with uncertain task execution time," in Proc. ACM DAC, 2007, pp. 664-669.

[15] Baliga, J, Ayre, R. W., Hinton, K., and Tucker, R. 2011. Green cloud computing: Balancing energy in processing, storage, and transport. Proceedings of the IEEE, 2011. 99(1):149–167.

[16] Canali, C. and Lancellotti, R. Parameter Tuning for Scalable Multi-Resource Server Consolidation in Cloud Systems. Communications Software and Systems, 2016.

[17] Urgaonkar, R., Kozat, U. C., Igarashi, K., and Neely, M. J. Dynamic resource allocation and power management in virtualized data centers. In NOMS, 2010, pp 479–486. IEEE.

[18] A. Alsarhan, A. Itradat, A. Y. Al-Dubai, A. Y. Zomaya and G. Min, "Adaptive Resource Allocation and Provisioning in Multi-Service Cloud Environments," in IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 1, pp. 31-42, Jan. 1 2018.

[19] P. Arroba, J. M. Moya, J. L. Ayala and R. Buyya, "DVFS-Aware Consolidation for Energy-Efficient Clouds," 2015 International Conference on Parallel Architecture and Compilation (PACT), San Francisco, CA, 2015, pp. 494-495.

[20] W. Chawarut and L. Woraphon, "Energy-aware and real-time service management in cloud computing," 2013 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Krabi, 2013, pp. 1-5.

[21] H. Faragardi, A. Rajabi, K. Sandström and T. Nolte, "EAICA: An energy-aware resource provisioning algorithm for Real-Time Cloud services," 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA), Berlin, 2016, pp.1-10.

[22] Z. Bagheri and K. Zamanifar, "Enhancing energy efficiency in resource allocation for real-time cloud services," Telecommunications (IST), 2014 7th International Symposium on, Tehran, 2014, pp. 701-706.

[23] Zheng, W., Sakellariou, R.: Budget-deadline constrained workflow planning for admission control. J. Grid Comput. 11(4), 105–119, 2013.

[24] T. Knauth and C. Fetzer, “Energy-aware scheduling for infrastructure clouds,” in Cloud Computing Technology and Science (CloudCom), 2012,IEEE 4th International Conference on. IEEE, 2012, pp. 58–65.

[25] S. K. Garg, C. S. Yeo, and R. Buyya, “Green cloud framework for improving carbon efficiency of clouds,” in Euro-Par 2011 Parallel Processing. Springer, 2011, pp. 491–502.

[26] Z. Zhu, G. Zhang, M. Li and X. Liu, "Evolutionary Multi-Objective Workflow Scheduling in Cloud," in IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 5, pp. 1344-1357, May 1 2016.

[27] D.B Srinivas, Puneeth R.P, Rajan M.A, Sanjay. H.A, An integrated framework to measure the energy consumption of a parallel application on a computational grid, International Journal of Computer Science and Network Security, VOL.16 No.7, pp. 94-98, 2016,