QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

WORKFLOW SCHEDULING IN COVID 19 LOCKDOWN CRISIS USING ENERGY AWARE MAKESPAN MINIMIZATION MECHANISM IN ARTIFICIAL INTELLIGENCE CLOUD COMPUTING

Rizhin Nuree Othman
Department of Computer Engineering, College of Computer Science and Engineering, Lebanese French University, Erbil, Kurdistan Region, Iraq
Rizhin.othman@lfu.edu.krd

ARTICLE INFO

Article History:
Received: 6/10/2023
Accepted: 23/11/2023
Published: Spring 2024

Keywords:
Energy, Consumption, Make span, task execution, EAMM

Doi:
10.25212/lfu.qzj.9.1.51

ABSTRACT

Artificial Intelligence Cloud computing provides the flexible and on-demand business environment based on the resource sharing phenomena to make the service easily available for public utility. Workflow scheduling has been one of the prominent cloud computing applications; workflow comprises repeated business activity pattern which needs to execute in accordance with sequential checklist scheduling and it requires efficient QoS such as energy consumption, task execution time. In this work we address the problem of makes pan minimization and energy consumption, We have developed an efficient workflow mechanism named EAMM (Energy Aware Makes pan Minimization) to achieve the better performance in workflow scheduling. At first EAMM mechanism designs the problem of processing delay and execution time as a joint problem and solves through the same algorithm. here we focus on minimizing the make span and energy consumption in VM scheduling this is achieved through reducing the execution time on given local processor through designed algorithm. EAMM is evaluated by considering the dataset of scientific workflow based Montage and through the comparative analysis it is observed that EAMM simply outperforms the existing model in terms of total execution time and energy consumption.

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

1. Introduction

Cloud computing is an emerging technology which makes use of internet and remote server for providing the services to users; cloud computing is considered as the novel paradigm where the business process and storage [1] which was applicable only for large organizations, can now easily accessed by the smaller companies and individuals as well. Moreover, cloud computing possesses various features such as on demand self-service, elasticity, pooling and broad network access; further self-service provides users to pay only for the amount of resource used and time period. Moreover, resource pooling indicates unlimited resource availability for customers and various customers can adopt various services in accordance with their demand; elasticity is an added advantage to scale up and scale down resources as per requirement. Virtualization is one of the primary ideas of cloud computing that allows various VMs to reside into the single machine [2]-[5]-[6].

Moreover, users use various VM instances to launch their application and later VM execute the task driven by user. Workflow comprises the repeatable business pattern activities; it is explained as the series of group, an organization, staff or operations represented in DAG (Directed Acyclic Graph) [7]. Here each graph node depicts the process or task and further edges indicates the task dependency. Moreover, executing the heavy workflows results in uncertainty error, hence workflows possess a huge challenge as sometimes hundred tasks need to be executed in such a way that time and cost has to be managed. Moreover, workflows execution is carried out in parts; it starts with submitting an application, making required input file accessible then data transfer and so on. Further there are other constraint which effects the workflow performance such as VMs failure, data transfer. Scheduling a workflow as mentioned is one of the most important and demanding tasks in cloud computing. It is an effortful task and is the primary step of execution of an application [8]. It is the deciding factor of performance of an application. Scheduling basically is a mapping of various workflow tasks on different VMs of various performance factors so as to achieve QoS of users [9]. A workflow is a representation of various independent business tasks

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

which are combined together and are contained in a flow using dependencies, which are a vital part of scheduling. However, workflow scheduling is an NP-hard problem in cloud computing which makes an optimal solution difficult to achieve [10]. Although there can be numerous objectives of users, the most common of them is time and cost. Since there can be numerous requirements (task) of the user with numerous cloud resources, scheduling is done so as to achieve the users' objectives of cost and time by proper allocation of tasks to resources [11].

There have been several methods for makespan minimization, one of the methods is known as DFS (Dynamic Frequency scaling) which achieves the makespan optimization and energy efficient scheduling through scaling down frequency and voltage when given tasks are running. Moreover, there are various mainstream manufacturers such as AMD, ARM and Intel which adopts the DVFS technology. Hence considering the importance of DVFS mechanism several important researches [12]-[13] included the DVFS mechanism for better performance.

This particular research work first section focusses on the parallel computation and the need for parallel computation and steps related to makespan optimization. Further the sub section of first section highlights the research contribution. Second section discusses the various existing methodologies used for the performance enhancement and their drawbacks are highlighted. Third section presents the EAMM mechanism along with the Makespan optimization model, resource allocation and task scheduling mechanism. Similarly, fourth section presents the evaluation of EAMM mechanism.

1.1 Motivation and contribution of this research work

In IaaS (Infrastructure as a cloud), users utilize the computing service based on their requirement and pay per use model, further it provides the scalable resources to execute the real-world application faster in case of application such as genetic science, earthquake analysis and astronomy. Moreover, workflow is general model to describe the scientific application where task set creates forms link among the nodes; further IaaS cloud uses the higher resources which requires high amount of energy and it takes

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

time to execute the task. Moreover, the task execution time also known as the makes pan is one of the key factors considering the sensitivity of the workflow application. Task execution time is one factor and another factor is energy consumption. As the energy consumption becomes more then it causes to be more costly and also affects the data center. Hence In this research work we address the problem of makes pan minimization and energy consumption. Further contribution of this research work can be highlighted through below points.

- In this research work, we have developed a mechanism named EAMM (Energy Aware makes pan minimization) mechanism to minimize makes pan and energy consumption.
- We develop mathematical formulation of EAMM and EAMM based algorithm to achieve the objective.
- Review analysis of existing efficient workflow model and highlight the drawbacks of existing workflow.
- In order to achieve the objective of makes pan minimization at first we consider the problem of minimizing the execution time and weight offloading. Later we minimize the ratio of performance and offline algorithm,.
- EAMM is evaluated by considering the standard scientific workflow montage and its variant which is in a form of DAX file. Further evaluation is carried out through comparative analysis.
- • Comparative analysis shows that our model excels in terms of energy consumption and task execution time.

Workflow execution is one of the important and vast areas of research; hence in order to design an efficient workflow model, we need to perform the extensive survey of existing model. In the next section we perform the extensive review of existing workflow model.

2. Literature Survey

In this section a brief literature is discussed to reduce the drawbacks such as energy consumption, ineffective resource allocation and performance degradation presented by different researchers in the field of cloud computing environment. A QoS aware

workflow scheduling technique algorithm was developed that identifies a workflow PCP and assigns that PCP to available resources that tends to reduce the resource cost to meet the deadlines. Further the algorithm implies a recursive approach for the previously scheduled task developed at PCP. Moreover, this algorithm also satisfies the general properties of cloud model such as flexibility and elasticity [14]. Moreover [15] developed cost aware workflow scheduling algorithm for reduction in makespan and workflow cost; it classifies the resources based on the different QoS workflow metrics. This algorithm was well intended to rule based mechanism for choosing the best fit workflows resources, this tries to optimize the workflow make span and meet the QoS constraint. [16] developed a partition-based strategy to solve the workflow scheduling problem, here algorithm develops the sub-workflow through the main workflow though partition strategy. This also further minimizes the task communication overhead and optimizes the execution time this strategy distributes the task on given optimal instances through maintaining the suitable scheduling strategy. [17] developed an efficient workflow scheduling mechanism through optimizing execution time, here this algorithm has two particular objectives i.e., meeting the budget constraint and minimizing the makespan. Here at first the strategy selects an optimal resource set which minimizes the total cost which meets the budget constraint later the algorithm tends to schedule the task in such optimal way that it suits the resource which is based on the heuristic approach. [18] designed dynamic workflow strategy application; in here algorithm computes the workflow execution and maintains the optimal scheduling strategy through dual stochastic function such as characteristics and distribution functions. Moreover, the primary intention here was to increase the estimating accuracy and workflow makespan to increase the algorithm performance. Moreover, algorithm applies different minimization to increase the algorithm performance and workflow scheduling strategy. [19] developed a dynamic workflow which was cost efficient and meets the deadline, this method considered the dynamic resource provisioning in cloud data center through mean clustering and subset sum problem. [20] developed a resource efficient workflow scheduling to meet the objective of makespan minimization and resource utilization, the author mainly

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

focused on the finding the optimal workflow scheduling such as server reliability maximization, reduction in total execution time and minimizing the makespan. [21] Developed a keen point driven algorithm to match up the objective, author focused on deadline aware and cost-efficient workflow scheduling to reduce the cost and makespan to meet the user defined constraint. Similarly [22] developed deadline aware cost-effective workflow scheduling to minimize makespan, here the algorithm tries to maintain the optimal scheduling plan for given workflows assigns the given task to the resources on given rank on rank-based policy. Other method like [23] and [24] focused on achieving the better efficiency model and achieve better performance, however it was mainly based on the cost and it comprises with scheduling performance.

In above literatures, various problems such as optimization problem, high computational complexity, energy consumption and ineffective resource utilization etc. exists which can degrade their performance and hence difficult to introduce in real time scenarios. Therefore, an effective resource scheduling technique is needed to maintain a balancing between energy consumption and high performance based on $DVFS$ due to its various energy saving capabilities for a cloud computing environment. Therefore, we have presented a EAMM technique for cloud computing devices which efficiently decreases energy consumption as well as executes operations in very less time through makespan minimization.

3. Proposed Methodology

In this work we develop and design a particular mechanism named EAMM (Energy Aware Makes pan minimization) mechanism to reduce the makes pan and improvise the energy efficiency of the overall model. Moreover, this is achieved through considering the delay in task processing;. In general a scheduler minimizes the makespan through unloading task, however it affects cost, considering these scenario we design EAMM mechanism. Our proposed model EAMM is tested on

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

$Montage$ scientific dataset; The Montage application is created by NASA/IPAC stitches together multiple input images to create custom mosaics of the sky [25]
This work was conducted using with operating system of windows 10 64-bit with 16 GB RAM and loaded with I5 processor; further the model 3.20 GHz CPU and the model is evaluated using the programming language using java and neon .3 editor.

4. System Model and Preliminaries

Let's consider any hybrid cloud model where the device access to m identical parallel process denoted by $j \in \mathcal{M} = \{1, \ldots\ldots, n\}$. Moreover, initially we consider that remote cloud as the single powerful processor referred as processor 0. Later we extend our work to multiple processors. Initially we assume that all tasks are available at time 0.

Let's consider non-pre-emptible and independent tasks n that are available to given scheduler at time is zero; let's consider $\mathcal{U} = \{1, \ldots, n\}$ be task indices with processing time for task is unknown and denoted as v_k. Moreover, main intention here is to optimize the makespan of scheduled task on the given processor; in here we consider the offloading cost and makes pan together through weighted sum. Let $\mathcal{T}$ be the set of possible schedules and t be the schedule, further t decides offloading of task on processor; let $\mathcal{U}_j(\mathcal{T})$ be the task scheduling set on processor $j \in \mathcal{M}U\{0\}$ under given schedule$t$. Consider $\mathcal{D}_j(t)$ as the total time taken to complete the assigned to processor and it is formulated through the below equation.

$$\mathcal{D}_j(t) = \sum_{k \in \mathcal{U}_j(t)} \mathcal{V}_k, \forall\, j \in \mathcal{M} \quad (1)$$

$$\mathcal{D}_0(t) = \sum_{k \in \mathcal{U}_0(t)} \Xi_k \quad (2)$$

Further as Ξ_k is unknown, cost is given as

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

$$C(t) = \sum_{j \in T_0(s)} \widehat{\mathcal{b}_{\hbar}} \quad (3)$$

Moreover total cost of schedule s is given through below equation

$$\aleph(t) \triangleq max_{j \in \mathcal{M} U\{0\}}\{\mathcal{D}_j(t)\} + x\gamma(t) \quad (4)$$

In the above equation w indicates weight parameter which allows tuning importance between cost and makespan, further cost minimization can be given as:

$$\underset{t \in \mathcal{T}}{minimize}\ \aleph(t) \quad (5)$$

5. Intermediate Framework

In here we design an intermediate framework where processing time $\mathcal{V}_{\hbar}$is unknown and costs are $\widehat{\mathcal{b}_{\hbar}}$, $\forall\ \hbar$; in order to develop this intermediate framework, we consider $\mathcal{Q}$ as a problem instance of $\mathcal{Q}_{sum}$.Further $t(\mathcal{Q})$ is schedule of online algorithm and $\bar{t}^*(\mathcal{Q})$ is schedule of an optimal algorithm. Further the interactive framework is given as:

$$\underset{\forall \mathcal{Q}}{max}\ \aleph(t(\mathcal{Q}))\left(\aleph(\bar{t}^*(\mathcal{Q}))\right)^{-1} \leq \delta \quad (6)$$

In here θ is tight for algorithm such that it satisfies the below equation.

$$\aleph(t(\mathcal{Q})) = \delta\aleph(\bar{t}^*(\mathcal{Q})) \quad (7)$$

6. Problem Definition

In this section we define the problem which is to reduce the makespan on given $m+1$ processor p_{max} as the cost of off-loading

$$\underset{t \in \mathcal{T}}{minimize}\ \mathcal{D}_{max}(t) \quad (8)$$

In the above equation $\mathcal{D}_{max}(t)$is formulated as

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

$$\mathcal{D}_{max}(t) \triangleq max\{max_{\in \mathcal{M}v\{0\}}\, \mathcal{D}_i\,(t), x\mathcal{U}(t)\} \quad (9)$$

Moreover q_{max} and $\mathcal{D}^*_{max}$ indicates objective and further optimal schedule is denoted throught^*; further if x=0 then q_{max} is minimal on $n+1$ processor.

6.1. Relation between q_{sum} and q_{max}

Let s' be considered as computed schedule for solving the q_{max}, further inequalities are formulated as:

$$\aleph(t') = \max_{j \in \mathcal{M} \cup \{0\}} \{\mathcal{D}_j(t')\} + x\mathrm{C}(t') \quad (10)$$

$$=2\delta\aleph(\bar{t}*)$$

Moreover in the above equation we observe that $q_{max}$and q_{sum}require similar solution, hence we develop mechanism for q_{max}. Moreover, this is achieved through establishing the lower bound for C^*_{max}

Let $\mathcal{D}^*_j$ denotes completion time and $\mathcal{U}^*_j$ indicates task scheduled on given processor j under optimal schedule $\mathcal{U}^*_j$.optimal equation is formulated as:

$$\mathcal{D}^*_j = \sum_{k \in \mathcal{U}^*_j} \forall_k \in \mathcal{M}, v_k \quad (11)$$

$$\mathcal{D}^*_0 = \sum_{k \in \mathcal{U}^*_0} \Xi v_k \quad (12)$$

$$\mathcal{D}^*_{max} \geq \sum_{k \in \mathcal{U}^*_j} \forall_j \in \mathcal{M}, v_k \quad (13)$$

$$\mathcal{D}^*_{max} \geq \sum_{k \in \mathcal{U}^*_j} \Xi v_k \quad (14)$$

Further substituting $\mathcal{D}^*_{max}$ in equation (12) and (13), we achieve:

$$\sum_{k=1} v_k \leq \left(n + \frac{1}{\Xi}\right) \mathcal{D}^*_{max} \quad (15)$$

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Further, $\mathcal{D}^*_{max}$ be the optimal objective, $\mathcal{t}'$ be the optimal schedule for scheduling task $\mathcal{U}$ with earlier assumption regarding the processing time. $\mathcal{D}^*_j$ be the schedule length, $\mathcal{U}'_0 \subseteq \mathcal{U}'$ be the task offloaded then we have further equation

$$\sum_{\mathcal{k}\in\mathcal{U}_0} \mathcal{b}_{\mathcal{k}} = \frac{1}{\varkappa}\sum_{j=1}^{n} \mathcal{D}^*_j + \sum_{\mathcal{k}\in\mathcal{U}'_0} \mathcal{b}_{\mathcal{k}} \quad (16)$$

Further we use$\mathcal{D}^*_{max} \geq \mathcal{D}^*_j,\ \forall j \in \mathcal{M}U\{0\}$ and $\mathcal{D}^*_{max} \geq \sum_{\mathcal{k}\in\mathcal{U}'_0} \mathcal{b}_{\mathcal{k}}$ and obtain EAMMM algorithm

$$\sum_{\mathcal{k}\in\mathcal{U}'} \mathcal{b}_{\mathcal{k}} \leq \left(1 + \frac{n}{\varkappa}\right) \mathcal{D}^*_{max} \quad (17)$$

6.2. EAMM Algorithm

EAMM algorithm forms a task lost in accordance with their offloading cost $\mathcal{b}_1$, further task in given list is scheduled one after the other on given processor $\mathcal{Y}$. Moreover, the proposed EAMM algorithm have not idea about the processing time of given task $\mathcal{k}$ on processor$\mathcal{M}$; hence $\varkappa\mathcal{b}_{\mathcal{k}}$ is used for computing the processing time of task $\mathcal{k}$ which is scheduled on processor and further task is terminated if it exceeds more than estimated time. Moreover, here we have considered $\varkappa \geq 1$ in such a way that task does not get terminated until it is processed.

Moreover after scheduling task and performing iteration; tasks that are terminated are sorted again and task lists are arranged in ascending order of $\mathcal{b}_k$; in next iteration task lists are scheduled where task are not terminated except last.

$Step1$:	$initialize\ m = 1, \mathcal{U}^{(m)} = 1$
$Step2$:	$While\ m <= 2\ do$
$Step3$:	$\mathcal{k}_1 = 1, \mathcal{k}_0 = \left\|\mathcal{U}^{(m)}\right\| + 1$

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Step4:	$Sort\ \mathcal{U}^{(m)}\ in\ ascending\ order\ of\ given\ \mathcal{b}_{\mathcal{k}}$
Step5:	$Re-indexing\ task\ in\ order\ of\ \mathcal{b}_1 \leq \mathcal{b}_2 \leq \cdots \leq \mathcal{b}_{\lvert\mathcal{U}^{(m)}\rvert}$
Step6:	$Starting\ of\ task\ \mathcal{k}_1\ on\ given\ processor\ \mathcal{Y}$
Step7:	$for\ \ell = 1\ to\ min\{n, \lvert\mathcal{U}^{(m)}\rvert\}\ do$
Step8:	$\mathcal{k}_0 = \mathcal{k}_0 - 1$
Step9:	$processing\ of\ task\ \mathcal{k}_0\ on\ given\ processor\ \ell$
Step10:	$If\ m = 1$ $Then$ $Terminate\ task\ \mathcal{k}_0$ if execution of task exceeds $\varkappa\mathcal{b}_{\mathcal{k}_0}$ and include in $\mathcal{U}^2$ $End\ if$ $End\ for\ loop$
Step11:	$While\ \mathcal{U}^{(m)} \neq 0\ do$
Step12:	$Waiting\ for\ event\ \mathcal{F}\ occurrence$
Step13:	$If\ \mathcal{F} = a\ task\ \hat{\mathcal{k}}$ (cancelled or completed on processor $\hat{\jmath} \in \mathcal{M}$)
Step14:	$Cancel\ the\ task\ \hat{\mathcal{k}}\ if\ it\ is\ scheduled$
Step15:	$\mathcal{U}^{(m)} = \mathcal{U}^{(m)} \backslash \{\hat{\mathcal{k}}\}$ $\mathcal{k}_0 = \mathcal{k}_0 - 1$
Step16:	$If(\mathcal{k}_0 > \mathcal{k}_1)\ then$

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

	Task scheduling on processor $\hat{\imath}$ End of if condition
Step17:	$If\ (m = 1)$ $Task\ cancellation\ of\ \hbar_0$ if the execution time exceeds $\varkappa\mathcal{b}_{\hbar_0}$ $Include\ task\ into\ \mathcal{U}^{(2)}$ $End\ of\ if\ statement$
Step18:	$Else\ if\ \mathcal{F}\ =\ task\ \hbar_1\ which\ is\ completed\ on\ processor\ \ then$ $Cancel\ task\ \hbar_1\ if\ particular\ task\ is\ scheduled\ on\ processor$
Step19:	$\mathcal{U}^{(m)} = \mathcal{U}^{(m)} \backslash \left\{\widehat{\mathcal{U}^{(m)}}\right\}$ $\hbar_0 = \hbar_0 - 1$
Step20:	$IF(task\ \hbar_1\ is\ not\ completed)\ then$ $Schedule\ task\ \hbar_1\ on\ processor$
$Step21$:	$End\ of\ if\ statement(step20)$
$Step22$:	$End\ of\ if\ stament(step13)$
$Step23$:	$End\ while\ (11)$
$Step24$:	$m = m + 1$
$Step25$:	$End\ while\ (step2)$

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

6.3. Minimizing the Ratio

In this section we tend to minimize the ratio of performance and offline algorithms. Here considering the set of tasks $\mathcal{U}^{(m)}$ in given m iteration. Schedule length is denoted by $\mathcal{D}_{max}^{(m)}$ (t^{EAMM}), t_m be the further schedule. The main advantage of proposed algorithm is that it can optimize the competitive ratio through proper $\varkappa$. In here, at first, we use higher value which allows task to run for longer period. Later we use smaller value of $\varkappa$ for aggressive cancellation. Further considering the optimization problem, here we tend to minimize the upper bound of ratio and minimization problem is given as:

$$\begin{matrix} minimize \\ \varkappa \geq 1 \end{matrix} \quad (18)$$

Further solutions of above equations are given as:

$$= \begin{cases} 1 & n = 1 \\ 0.5(n)^{1/2} & n \geq 2 \end{cases} \quad (19)$$

Further using the above equation of $\mathcal{D}_{max}$ we get

$$(t^{EAMM})(\mathcal{D}_{max}^{*})^{-1}\mathcal{D}_{max} \leq \begin{cases} 4 & n = 1 \\ 1 + 2\,(2n)^{1/2} & n \geq 2 \end{cases} \quad (20)$$

7. Result & Discussion

Now a days, the request of CC (cloud computing) resources has highly emerged in real-time due to its vibrant uses, flexibility, cost effective and easily accessible at anywhere anytime through internet. Multimedia-signal-processing method is well-known technique that can be utilized in these CC-devices. Therefore, the performance of these computing devices must be superior due to the extensive demand of these computing devices in day-to-day life. However, high energy consumption in these computing devices can disturb their performance; further makespan is an important constraint, hence to optimize these objectives, we have introduced EAMM

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

For heterogeneous computing devices which efficiently reduce energy consumption as well as provide superior performance. The run-time can be evaluated considering various jobs as 25, 50, 100, and 1000. Graphical representation of our outcomes is also presented considering execution-time, number of tasks and energy consumption. The run-time and total power consumed can be evaluated using different parameters in table 1 which is demonstrated in the following section. Figure 1 shows the montage scientific workflow.

EAMM is evaluated through varying the number of virtual machines as 20, 40 and 60, these varied results are compared with the existing model by considering the for eminent parameter i.e. total execution time, power sum, power average, average power and energy consumption. In here table 2 and table 3 presents the comparison of existing model with the proposed EAMM model by varying the virtual machine as 20, 40 and 60.

Table (1): Comparison of existing model and EAMM on virtual machine 20

	ES			EAMM		
Number of VM	20	*40*	60	20	40	60
Total Execution Time (s)	6359.41 sec	12380.99 sec	24712.57 sec	3030.25 sec	2953.86	5898.82 sec
Energy Consumption (Wh)	3495.42	7028.52	3495.42	416.88	1330.92	894.063

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Table (2): Comparison of existing and EAMM on virtual machine 40

	ES			EAMM		
Number of VM	20	*40*	60	20	40	60
Total Execution Time (s)	6359.41 sec	12380.99 sec	24712.57 sec	3030.25 sec	2953.86	5898.82 sec
Energy Consumption (Wh)	3495.42	7028.52	3495.42	416.88	1330.92	894.063

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

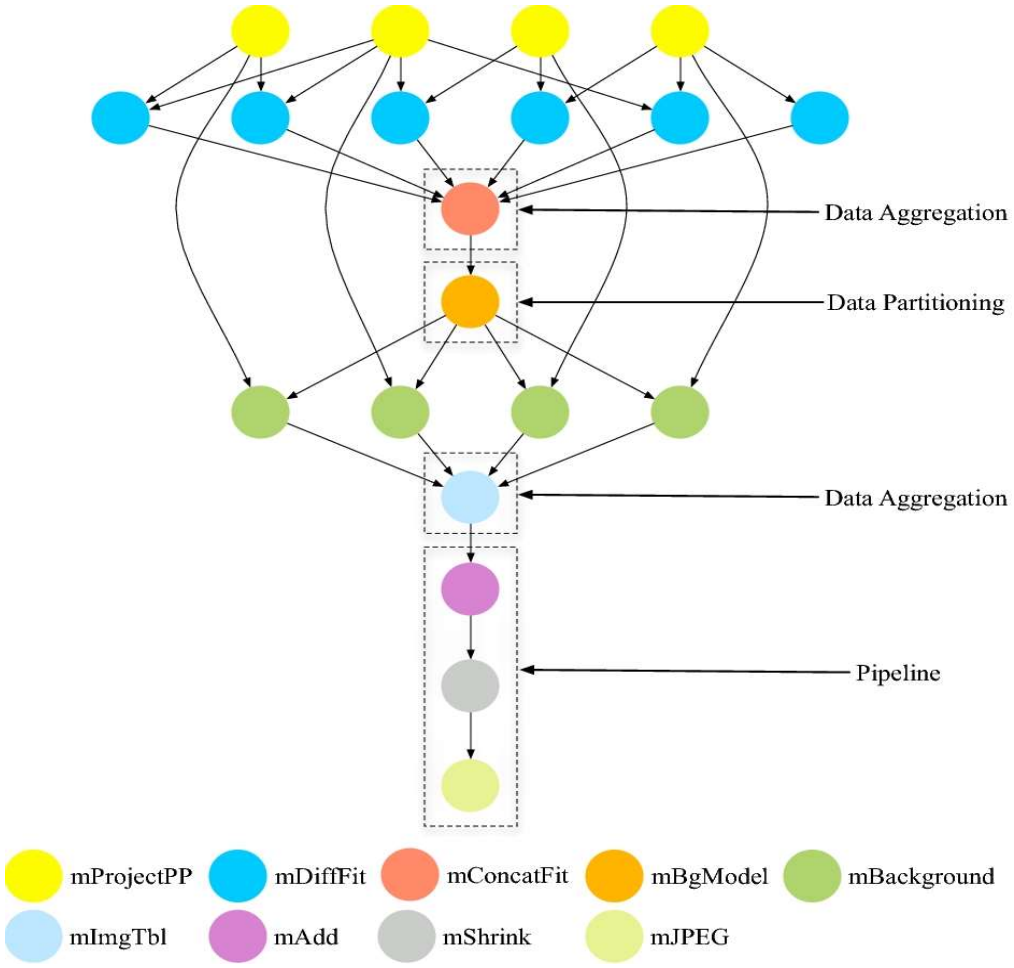


Figure (1): Montage Scientific Workflow

7.1 Montage_25

In this sub-section we evaluate the EAMM methodology on montage_25 node dax by varying the number of VM; further comparison is given based on two parameters i.e., total time taken and energy consumption as both are very important factor. In here below figure i.e., figure 2 and figure 3 shows comparison on total simulation time and Energy Consumption. In figure 2, X-Axis indicates number of VM, y- axis indicates time taken in seconds. Moreover, considering the dynamic environment of cloud the less

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

time it takes the better and efficient the model; in here for 20 VM existing model takes 6310.87 seconds whereas EAMM mechanism takes only 2883.65 seconds. For 40 VM time taken to complete the task is 8965.28 and for proposed time taken is 2883.65. Similarly for 60 VM existing model takes 6923.13 sec and EAMM takes 2883.65 sec. Further for 20 VM, 40VM and 60 VM energy consumption for existing mechanism is 3491.37, 5423.21 and 4079.23 respectively whereas EAMM mechanism takes 1280.94, 1287.94 and 1287.94 respectively.

Table (3): Tabular Comparison of Existing Model and EAMM on M ontage_25

	ES			EAMM		
Number of VM	20	*40*	60	20	40	60
Total Execution Time (s)	6310.87 sec	8965.28 sec	24712.57 sec	2883.65 sec	2883.65	2883.65sec
Energy Consumption (Wh)	3491.37	5423.21	4079. 23	1280.94	1287.94	1287.94

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

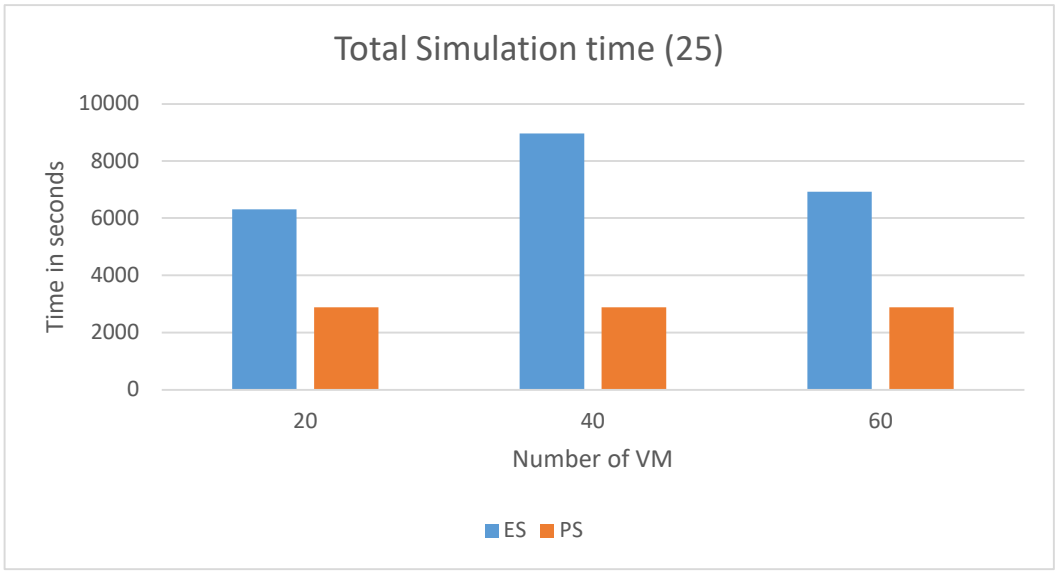


Figure (2): execution time comparison on maontage 25

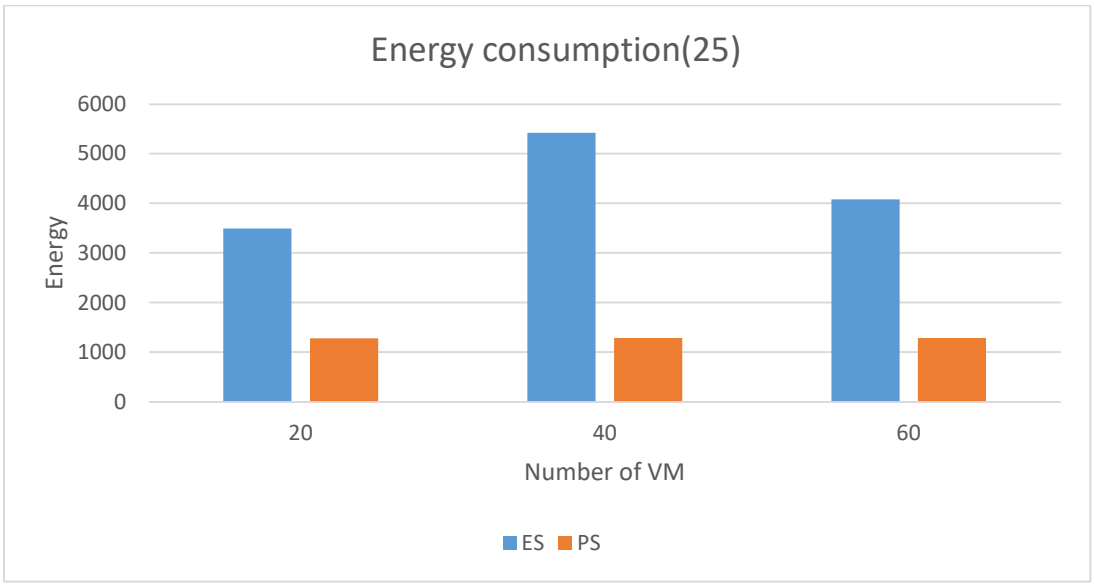


Figure (3): Energy Efficientcomparison on maontage 25

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Further evaluation is carried out on Montage _50; in here for 20 VM, simulation time taken is 9272.74, 8965.28 and 6923.28 respectively whereas proposed mechanism takes 2984.39 sec in case of all VM. Hence it is observed that proposed model takes comparatively less time than the existing mechanism. Similarly, energy consumption for VM 20, 40 and 60 are 5649.82, 5423.21 and 4079.23 respectively; however proposed mechanism consumes 1340.94, 1456.94 and 1480.94 respectively.

Table (4): tabular comparison of existing model and EAMM on Montage_50

	ES			EAMM		
Number of VM	20	*40*	60	20	40	60
Total Execution Time (s)	9272.74 sec	8965.28 sec	6923.28 sec	2984.39 sec	2984.39	2984.39 sec
Energy Consumption (Wh)	5649.82	5649.82	4079.23	1340.94	1456.94	1480.94

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

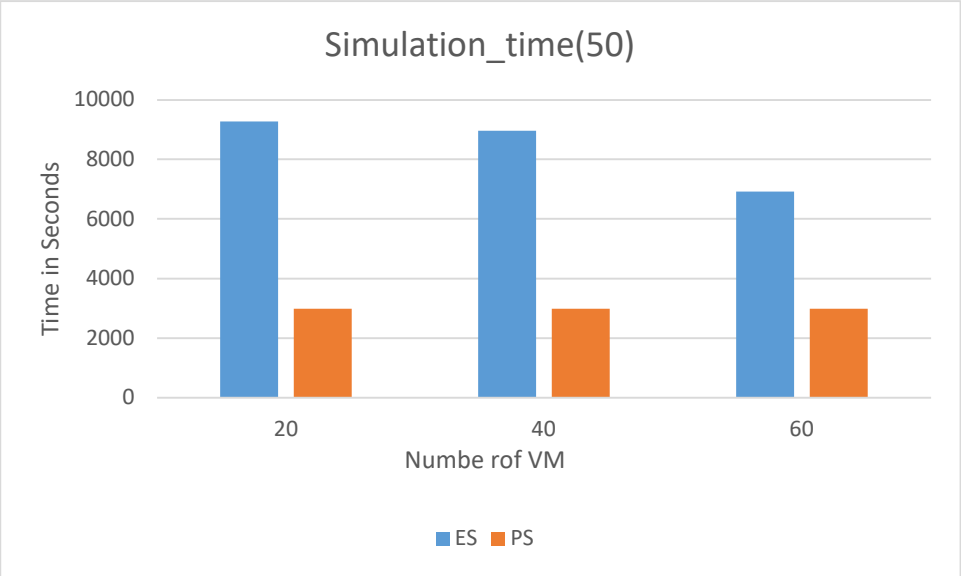


Figure (4): Graphical Comparison of Existing Model and Proposed Model in Montage_50 Based on Execution Time

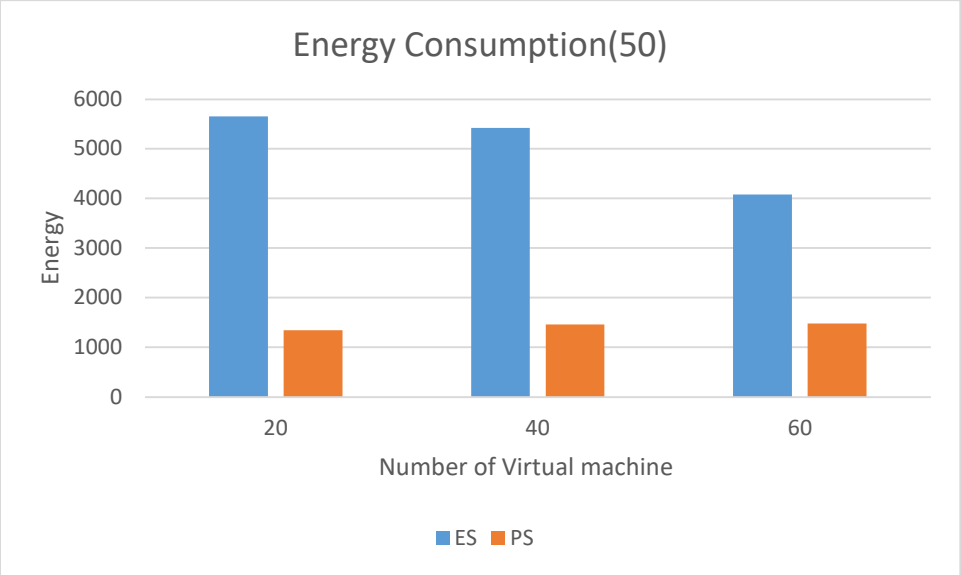


Figure (5): Energy Efficient Comparison on Mantage 50

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

7.2 Montage_100

Table 5 shows the comparative analysis of existing and proposed mechanism on Montage 100 node dax, here the simulation time taken through 20, 40 and 60 for existing model are 8765.47, 12217.37 and 12737.5 respectively whereas proposed mechanism takes Moreover energy consumption for montage 100 through 20, 40 and 60 are 6313.95, 8501.44 and 8867.34 whereas EAMM mechanism consumes 3196.13, 3171.32 and 3171.32 respectively.

Table (5): Comparison of Existing and EAMM on Montage 100

	ES			EAMM		
Number of VM	20	*40*	60	20	40	60
Total Execution Time (s)	8765.47sec	12217.37 sec	12737.5 sec	6313.95sec	8501.44	8867.34 sec
Energy Consumption (Wh)	6313.95	8501.44	8867.34	3196.13	3171.32	3171.32

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

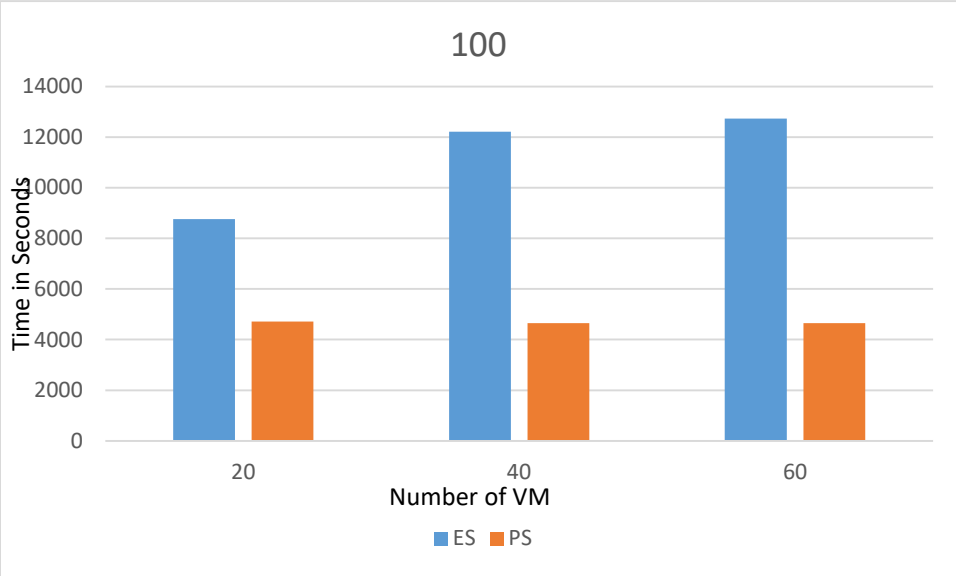
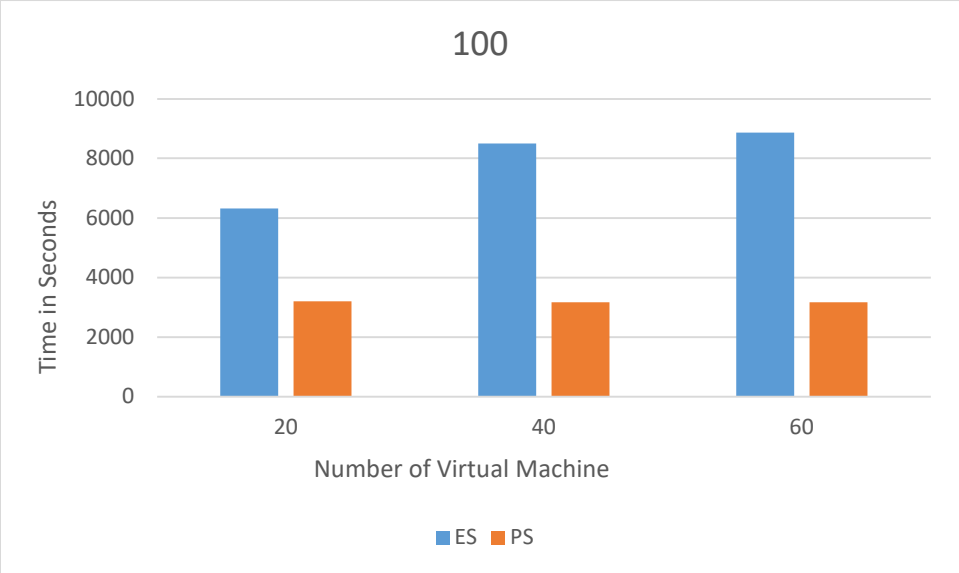


Figure (5): Task Execution Time Comparison on Montage 100



QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Figure (6) Energy Efficient Comparison on Montage 50

7.3 Montage_1000

Table 6 shows the comparison of existing and EAMM mechanism on Montage 1000 node DAX, in here for Montage 1000 through 20, 40 and 60 VM existing model takes 8765.47, 12217.74 and 12737.5 respectively to execute the task whereas EAMM takes 5715.3, 5646.85 and 5646.85 respectively. Further Energy consumed through 20, 40 and 60 are 6313.95, 8501.44 and 8867.34 respectively for existing model, whereas 4196.13, 4171.32 and 4171.32 respectively for EAMM mechanism.

Table(6): Comparison of Existing and EAMM on Montage 100

	ES			EAMM		
Number of VM	20	*40*	60	20	40	60
Total Execution Time (s)	8765. 47sec	12217.74 sec	12737.5 sec	5715.3sec	5646.85	5646.85 sec
Energy Consumption (Wh)	6313.95	8501.44	8867.34	4196.13	4171.32	4171.32

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

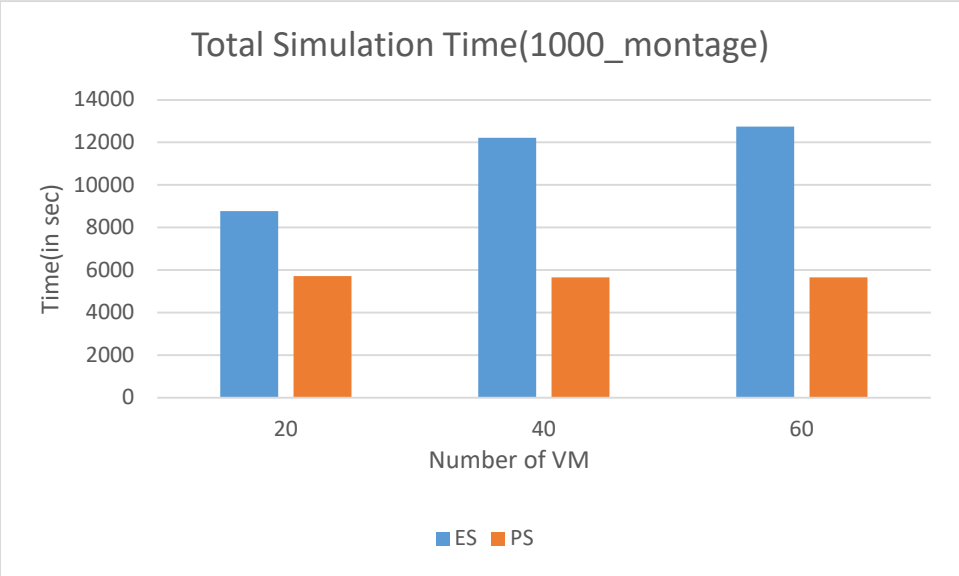
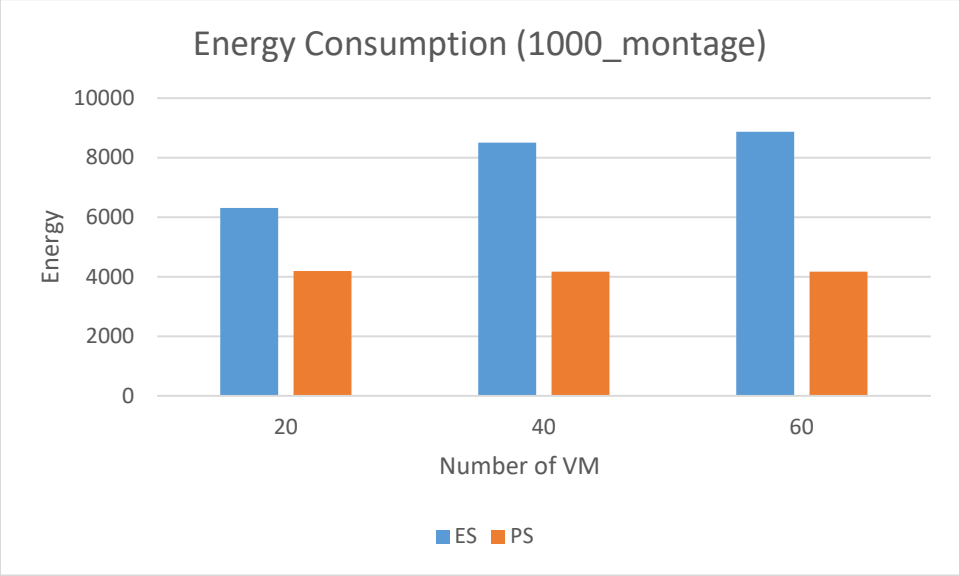


Figure (7): Task Execution Time Comparison on Montage 1000



QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Figure (8): Energy Consumption Comparison on Montage 1000

8. Conclusion

Workflow scheduling has become one of the important and challenging issues considering the emerging distributed environment; further due to the dynamic environments it's essential to satisfy the QoS constraint such as Energy consumption, makes pan. Cloud gets the application in form of workflow that comprises the inter-dependent task set for solving the enterprise or large-scale scientific problem. In this research work we focus on minimizing the makes pan through monitoring the processing delay. Further EAMM is evaluated considering the important metrics as makes pan and energy consumption; in order to evaluate we consider the scientific workflow named montage and its various model; further evaluation is considered through comparative analysis with the existing model of makes pan and energy consumption and through the comparative analysis it is observed that our model simply outperforms the existing model.

Workflow execution is an extensive research area with various numbers of constraint and parameter, although EAMM shows the remarkable improvement in terms of task execution time and energy consumption. There are several research areas such as power aware which we will be focusing in the next research work and evaluating model by considering the different workflow to prove the better efficiency of model.

REFERENCE

1. E. M. Mocanu, M. Florea, M. I. Andreica and N. Ţăpuş, "Cloud Computing—Task scheduling based on genetic algorithms," *2012 IEEE International Systems Conference SysCon 2012*, Vancouver, BC, 2012, pp. 1-6, doi: 10.1109/SysCon.2012.6189509.
2. G. Junwei, S. Shuo and F. Yiqiu, "Cloud resource scheduling algorithm based on improved LDW particle swarm optimization algorithm," 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, 2017, pp. 669-674.
3. M. Shelar, S. Sane, V. Kharat and R. Jadhav, "Autonomic and energy-aware resource allocation for efficient management of cloud data centre," 2017 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, 2017, pp. 1-8.
4. A. Malatpure, F. Qadri and J. Haskin, "Experience Report: Testing Private Cloud Reliability Using a Public Cloud Validation SaaS," 2017 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), Toulouse, 2017, pp. 56-56.
5. Chung-Yiwu, H. Y. Yu, J. C. Huang and J. J. Chen, "A hierarchical reliability-driven scheduling for cloud video transcoding," 2015 International Conference on Machine Learning and Cybernetics (ICMLC), Guangzhou, 2015, pp. 456-461.
6. Pourbahrami, S., Balafar, M. A., Khanli, L. M., & Kakarash, Z. A. (2020). A survey of neighborhood construction algorithms for clustering and classifying data points. Computer Science Review, 38, 100315.
7. X. J. Xu, C. B. Xiao, G. Z. Tian and T. Sun, "Hybrid Scheduling Deadline-Constrained Multi-DAGs Based on Reverse HEFT," 2016 International Conference on Information System and Artificial Intelligence (ISAI), Hong Kong, 2016, pp. 196-202.
8. J. Singh, S. Betha, B. Mangipudi, and N. Auluck, “Contention aware energy efficient scheduling on heterogeneous multiprocessors,” IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 5, pp. 1251–1264, May 2015.
9. A. Zhou, S. Wang, B. Cheng, and Z. Zheng, “Cloud service reliability enhancement via virtual machine placement optimization,” IEEE Trans. Serv. Comput., pp. 1–1, Jan. 2016.
10. Z. Cai, X. Li, and J. N. D. Gupta, “Heuristics for provisioning services to workflows in xaas clouds,” IEEE Trans. Serv. Comput., vol. 9, no. 2, pp. 250–263, Mar.-Apr. 2016.

11. Z. Tang, L. Qi, Z. Cheng, K. Li, S. U. Khan, and K. Li, “An energy-efficient task scheduling algorithm in dvfs-enabled cloud environment,” J Grid Comput., vol. 14, no. 1, pp. 55–74, Mar. 2016
12. H. Chen, J. Wen, W. Pedrycz and G. Wu, "Big Data Processing Workflows Oriented Real-Time Scheduling Algorithm using Task-Duplication in Geo-Distributed Clouds," in *IEEE Transactions on Big Data*, vol. 6, no. 1, pp. 131-144, 1 March 2020, doi: 10.1109/TBDATA.2018.2874469.
13. Y. Kong, M. Zhang, and D. Ye, “A belief propagation-based method for task allocation in open and dynamic cloud environments,” Knowl.-Based Syst., vol. 115, pp. 123–132, Jan. 2017.
14. Ehab Nabiel Alkhanak, Sai Peck Lee, and Saif Ur Rehman Khan. 2015. Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities. *Fut. Gen. Comp. Syst.* 50 (2015), 3–21.
15. Toktam Ghafarian and Bahman Javadi. 2015. Cloud-aware data intensive workflow scheduling on volunteer computing systems. Fut. Gen. Comp. Syst. 51 (2015), 87–97.
16. Weihong Chen, Guoqi Xie, Renfa Li, Yang Bai, Chunnian Fan, and Keqin Li. 2017. Efficient task scheduling for budget constrained parallel applications on heterogeneous cloud computing systems. Fut. Gen. Comp. Syst. 74 (2017), 1–11.
17. Artem M. Chirkin, Adam S. Z. Belloum, Sergey V. Kovalchuk, Marc X. Makkes, Mikhail A. Melnik, Alexander A.Visheratin, and Denis A. Nasonov. 2017. Execution time estimation for workflow scheduling. Fut. Gen. Comp. Syst.75 (2017), 376–387.
18. Vishakha Singh, Indrajeet Gupta, and Prasanta K. Jana. 2018. A novel cost-efficient approach for deadline constrained workflow scheduling by dynamic provisioning of resources. Fut. Gen. Comp. Syst. 79 (2018), 95–110.
19. Young Choon Lee, Hyuck Han, Albert Y. Zomaya, and Mazin Yousif. 2015. Resource-efficient workflow scheduling
    a. in clouds. *Knowl.-Based Syst.* 80 (2015), 153–162.
20. Xin Ye, Sihao Liu, Yanli Yin, and Yaochu Jin. 2017. User-oriented many-objective cloud workflow scheduling based on an improved knee point driven evolutionary algorithm. *Knowl.-Based Syst.* 135 (2017), 113–124.

21. Raza Abbas Haidri, Chittaranjan Padmanabh Katti, and Prem Chandra Saxena. 2017. Cost effective deadline aware scheduling strategy for workflow applications on virtual machines in cloud computing. *J. King Saud Univ. Comput. Inf. Sci.* (2017). DOI:https://doi.org/10.1016/j.jksuci.2017.10.009.
22. Klavdiya Bochenina, Nikolay Butakov, and Alexander Boukhanovsky. 2016. Static scheduling of multiple workflows with soft deadlines in non-dedicated heterogeneous environments. *Fut. Gen. Comp. Syst.* 55 (2016), 51–61.
23. H. R. Faragardi, M. R. Saleh Sedghpour, S. Fazliahmadi, T. Fahringer and N. Rasouli, "GRP-HEFT: A Budget-Constrained Resource Provisioning Scheme for Workflow Scheduling in IaaS Clouds," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 6, pp. 1239-1254, 1 June 2020, doi: 10.1109/TPDS.2019.2961098.
24. Kakarash, Z. A., Karim, S. H. T., Ahmed, N. F., & Omar, G. A. (2021). New Topology Control base on Ant Colony Algorithm in Optimization of Wireless Sensor Network. Passer Journal of Basic and Applied Sciences, 3(2), 123-129.
25. Pourbahrami, S., Balafar, M. A., Khanli, L. M., & Kakarash, Z. A. (2020). A survey of neighborhood construction algorithms for clustering and classifying data points. Computer Science Review, 38, 100315.

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

بەردەوامبوونی کار لە قەیرانی قفڵدانی کۆڤید 19 بە بەکارهێنانی میکانیزمی وزەی ئاگادارکەرەوە لە کەمکردنەوەی ئەو ماوەیەی کە لە سەرەتای کارکردنەوە تا کۆتایی پێویستە لە زیرەکی دەست کردی هەوری کۆمپیوتەری

پـوخـتـه:

ژمێریاری هەوری زیرەکی دەستکرد ژینگەیەکی کاری نەرم و لەسەر داواکاری دابین دەکات لەسەر بنەمای دیاردەی هاوبەشکردنی سەرچاوە بۆ ئەوەی خزمەتگوزاریەکە بە ئاسانی بەردەست بێت بۆ سوودی گشتی. زیاتر لەم سااڵنەی دواییدا، خشتەی کاری کار یەکێک بووە لە بەرنامە دیارەکانی رایانکردنی هەور لێشاوی کار قاڵبی دووبارەی چالاکی بازرگانی پێکدێنێت کە پێویستە جێبەجێ بکرێت بەپێی لیستی پشکنینی زنجیرەیی. زیاتر لە کاتی خشتەی بەردەوامبوونی کار پێویستی بە QoSکوالیتی خزمەتگوزاری کارا هەیە وەک بەکارهێنانی وزە، کاتی جێبەجێکردنی ئەرکەکان پارامیتەری گرنگن؛ لە ڕابردوودا چەندین توێژەر سەرنجیان خستە سەر بەدەستهێنانی ئەدای باشتر، هەرچەندە کەم و کوڕی سەرەکی ئەم مۆدێلانە بە کارامەییانەوە دەبێ.

بۆیە، لەم توێژینەوەیەدا میکانیزمی وزەی ئاگادارکەرەوە لە کەمکردنەوەی ئەو ماوەیەی کە لە سەرەتای کارکردنەوە تا کۆتایی پێویستە کارپێکردوو بە ناوی EAMM پەرەپێدەدەین بۆ بەدەستهێنانی باشتری ئەدا لە خشتەی کار. لە سەرەتادا میکانیزمی EAMM کێشەی پرۆسەی دواخستن و کاتی جێبەجێکردن وەک کێشەیەکی هاوبەش دیزاین دەکات و لە ڕێگەی هەمان لۆگاریتمەوە چارەسەر دەکات. زیاتر ئەم توێژینەوەیە کارەکە جەخت دەکاتەوە لەسەر بچووککردنەوەی ماوەی و بەکارهێنانی وزە لە خشتەی ئامێری مەجازی . ئەمە زیاتر بەدەست هاتووە لە ڕێگەی کەمکردنەوەی کاتی جێبەجێکردن لەسەر سازکەری ناوخۆیی دراوە لە ڕێگەی لۆگاریتمی دیزاینکراو. EAMM زیاتر هەڵدەسەنگێنرێت بە لەبەرچاوگرتنی لێشاوی کاری داتای سێتی زانستی لەسەر بنەمای مۆنتاژ و لە ڕێگەی شیکردنەوەی بەراوردکارییەوە ئەوە تێبینی دەکرێت کە EAMM بە سادەیی مۆدێلی بەردەست لە ڕووی کاتی تەواو جێبەجێکردن و بەکارهێنانی وزە دەردەکات.

QALAAI ZANISTSCIENTIFIC JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil, Kurdistan, Iraq
Vol. (9), No (1), Spring 2024
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

جدولة سير الاعمال فى ازمة كوفيد 19 باستخدام الية تقليل الوقت التنفيذ فى الحوسبة السحابية فى الذكاء الاصطناعى

الملخص:
توفر الحوسبة السحابية للذكاء الاصطناعي بيئة عمل مرنة عند الطلب استنادا إلى ظواهر مشاركة الموارد لجعل الخدمة متاحة بسهولة للمنفعة العامة. وعلاوة على ذلك، في السنوات الأخيرة، كانت جدولة سير العمل واحدة من تطبيقات الحوسبة السحابية البارزة. يتضمن سير العمل نمط النشاط التجاري المتكرر الذي يحتاج إلى التنفيذ وفقا لقائمة التحقق التسلسلية. وعلاوة على ذلك جدولة سير العمل يتطلب كفاءة جودة الخدمة مثل استهلاك الطاقة، وقت تنفيذ المهمة هي المعلمة الهامة. في الماضي ركز العديد من الباحثين على تحقيق أداء أفضل، ولكن العيب الرئيسي لهذه النموذج يكمن في كفاءته. في هذا البحث عملنا على تطوير آلية فعالة لسير العمل اسمها EAMM (تقليل الوقت اللازم لبدء و انهاء مهمة بادراك الطاقة) لتحقيق أداء أفضل في جدولة سير العمل.

في البداية آلية EAMM تصمم مشكلة تأخير المعالجة ووقت التنفيذ كمشكلة مشتركة ويحل من خلال نفس الخوارزمية. كذلك يركز هذا العمل البحثي على تقليل فترة التصنيع واستهلاك الطاقة في جدولة الماكينة الذكية VM. وعلاوة على ذلك يتم تحقيق ذلك من خلال تقليل وقت التنفيذ على معالج محلي معين من خلال خوارزمية مصممة. كذلك يتم تقييم EAMM من خلال النظر في مجموعة البيانات من سير العمل العلمي القائم على المونتاج ومن خلال التحليل المقارن لوحظ أن EAMM يتفوق ببساطة على النموذج الحالي من حيث إجمالي وقت التنفيذ واستهلاك الطاقة.