A Combined Traffic and Workload-aware Optimized Virtual Machine Migration in Cloud Computing

Gowriprakash R[1]*, Shankar R[2], Duraisamy S[3]

[1]Ph.D., Research Scholar, Department of Computer Science, Chikkanna Government Arts College, Tiruppur, Tamilnadu, India
[2,3]Assistant Professor, Department of Computer Science, Chikkanna Government Arts College, Tiruppur, Tamilnadu, India
*Corresponding author's Email: prakashcs003@gmail.com

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021

Abstract: In cloud computing paradigm, Virtual Machine (VM) migration strategies play a vital role in reducing the Energy Consumption (EC) and balancing the workloads in the cloud servers. From this viewpoint, an Osmotic Hybrid artificial Bee and Ant Colony with Future Utilization Prediction and Multipath Traffic Routing (OH-BAC-FUP-MTR) strategy was designed to accomplish effective load-balancing and minimize the chance of traffic congestion during VM migration in data centres. But, the tradeoffs among workload efficiency and power-gain in heterogeneous cloud servers were not analyzed. Hence in this article, a VM consolidation strategy is proposed with OH-BAC-FUP-MTR to switch the idle Physical Machines (PMs) into hibernation mode, resulting in very less EC. In this strategy, the VM migration is targeted at consolidating the VMs depending on the workload to the smaller number of PMs for reducing the power use and encouraging green computing. Initially, the PMs are split into various groups depending on their workload levels and then a new Merge-and-Split-based Coalitional Game-theoretic (MSCG) method is applied to select associates from these groups to create efficient coalitions. After that, OH-BAC-FUP-MTR is performed amongst the coalition associate for maximizing the reward of each coalition and PMs are maintained to operate in the maximum power-efficient condition. Finally, the investigational outcomes exhibit the OH-BAC-FUP-MTR-MSCG achieves a mean EC of 60.63KWh which is 28.87% less than all other classical VM migration strategies.

Keywords: Cloud computing, Load balancing, traffic congestion, OH-BAC-FUP-MTR, VM consolidation, Workload aware, Coalitional game.

1. Introduction

Generally, the cloud computing model has many challenges because its deployment is rapidly rising. Handling loads between resources is the most critical issue in the cloud model. To address this issue, load-balancing strategies have been applied which boosts the total resource utilization, reliability, suitability and other properties in the cloud servers. Load-balancing is mainly used to manage the workload instability between the cloud servers for avoiding over-loaded and under-loaded conditions. By maintaining the Service Level Agreement (SLA) and client satisfaction, the load-balancing strategies are improved. Consequently, the use of effective load-balancing applications is a major part of cloud computing. So, various strategies are designed for load-balancing and task allocation tasks in data centres [1-4].

In these days, dramatic growth has focused at an extended version in osmotic computing preceded by the notion of the substance osmotic traits. Normally, it is used to maintain a sustainable use of resources in large-scale distributed computing [5-6]. In cloud platforms, it is applied to allow the usage of balanced VMs which are migrated in the edge devices. In many load-balancing strategies, though different optimization algorithms realize superior efficiency, many of them have no ability to increase the efficiency in every characteristic. As a result, an OH-BAC strategy [7] has been developed to lessen the power usage, more VM migrations and the host's turn-off. But, it reduces only few amounts of active PMs depending on their present resource needs whereas it omits the future resource needs. Therefore, the redundant VM migrations have been occurred and the rate of SLA Violations (SLAV) was increased in the server storage.

Thus, an OH-BAC-FUP strategy has been designed [8] to decrease the number of VM migrations and improve the load-balancing efficacy. In OH-BAC-FUP, the future and the current utilization of resources were estimated using to shift the VMs to the fewer amounts of effective PMs. The considered resources were CPU, bandwidth, storage size and memory of both VMs and PMs. These estimated values were given to the OH-BAC's objective function to choose the most apt PM. On the contrary, it has high chance to occur the traffic congestion if the bandwidth use was high between VMs within the cloud server. The ineffective localization of VMs tends to inter-VM transfer for linking bottleneck system links resulting in high traffic congestion. Additionally, the major node over-provisioning and uneven allocations in cloud servers give long-lasting traffic.

When there was a high amount of traffics, the resource use efficiency was rapidly degraded. Several researches focused on migrating the VMs with its influence on traffic; yet, the key intent was to decrease the total power use in a cloud server. Since cloud servers send a huge number of traffic, link breaks have severe effects [9]. While resources were engaged with sufficient backup bandwidth resources, 100% traffic avoidance was established. Traffic was avoided partly since the backup resources reservation was costly.

Even though partial avoidance may ensure ease of access, smaller bandwidth and efficacy were satisfactory for several uses. From this viewpoint, an OH-BAC-FUP-MTR strategy has been developed for establishing efficient load-balancing and minimizing the chance of traffic congestion in cloud servers [10]. In this strategy, the flow was transferred on many link-disjoint paths to avoid the breakages and guarantee the accessibility of at least single path for a flow upon path breakage or congestion. If congestion happens during migrating VMs to the PMs, then the MTR protocol was conducted which splits traffic into many types and transmits them through multiple link-disjoint paths. Additionally, the maximum flow on the link was considered as a congestion value. So, the flow was minimized if ensuring the bandwidth and avoidance grade needs. Conversely, the tradeoffs among workload efficiency and power-gain in heterogeneous cloud servers were not analyzed.

Therefore in this paper, a VM consolidation strategy is proposed with OH-BAC-FUP-MTR to switch the idle PMs into hibernation mode, resulting in very less EC. First, the challenges pertaining to storage elements in the cloud servers are investigated. Then, a unique classification was performed for guaranteeing the balanced workload during distribution and the major goal is on the VM migration strategy. The VM migration is targeted at consolidating the VMs depending on the workload to the smaller number of PMs for reducing the power use and encouraging green computing. Initially, the PMs are split into various groups depending on their workload levels and then a MSCG method is applied to select associates from these groups to create efficient coalitions. Then, OH-BAC-FUP-MTR is performed between the coalition associates for maximizing the reward of each coalition and PMs are maintained to operate in a high energy-efficient state. Thus, the tradeoffs between workload fairness and performance efficiency are balanced effectively.

The rest of the paper is planned as: Section II studies the previous works associated with the workload-aware based VM migration. Section III explains the OH-BAC-FUP-MTR-MSCG and Section IV displays its simulation findings. Section V summarizes the work and recommends the future development.

2. Literature survey

Liu et al. [11] developed a novel VM migration method depending on the cloud model time series workload prediction technique. According to the configuration of maximum and minimum workload limits for host machines, monitoring the tendency of their successive workloads via generating the workload time series using the cloud model and stipulating the normal VM migration condition workload-aware migration, an origin and target host as well as a VM on the origin host were chosen. But, its error rate was still high and has and does not consider the traffic on the link during migration.

De Maio et al. [12] introduced a workload-aware energy usage framework to migrate the VMs. The major objective of this framework was to increase the accuracy of many previously omitted parameters during VM migration. Initially, the parameters considered in migrations were addressed. After, their probable effect on data centre level energy use was analyzed when accounting various workloads. But, it does not consider the effect of the network-intensive workloads on the target host.

Arroba et al. [13] designed a Dynamic Voltage and Frequency Scaling (DVFS) strategy which minimizes the power use when mitigating the efficiency degradation. Also, a DVFS-aware consolidation strategy was developed which optimizes the consumption by configuring the DVFS. The tradeoffs between energy use and efficiency degradation was considered to maintain the QoS. But, it does not consider the effect of workload.

Guo et al. [14] presented a game-based consolidation technique of VMs with power and load constraints. Initially, each measured value of the resource load was analyzed by the t-test for removing outliers. Then, the future resource load was monitored by the gray theory. Also, all online PMs were combined by the number of VMs on them and their future load values. According to the combinations, a pre-processing method was applied to choose target PMs and calculate the group of target PMs for a VM awaiting migration. At last, the final target PM for the VM was chosen by the game-based techniques intended at optimizing the total energy use. But, it does not consider the VM's relevance. Also, it needs to adjust the migration cost and increase the prediction accuracy when using continuous real-time workload.

Wu et al. [15] developed an intelligent energy usage framework for VMs in which the energy signatures of VMs were initially analyzed in various settings through analysis. After, a VM energy model called CAM was designed to adapt the resetting of VMs and provide precise energy estimation under CPU-intensive workload. Additionally, different training methods were developed related to the common conditions for model training. But, it needs to estimate the energy of nodes operating Input/Output (I/O)-intensive and data-oriented purposes.

Li et al. [16] developed an Energy-efficient and Quality-aware VM Consolidation (EQ-VMC) technique. First, they observed that VM localization was a secure hybrid optimization problem with several resource constraints. After, the possible mappings between VMs and PMs were abstracted as a part of constrained search space and which related to the population of heuristic evolutionary algorithm. Every individual of population was equivalent to a real mapping between VMs and PMs during a cycle of VMs consolidation. After, a hybrid optimization using an improved heuristic evolutionary algorithm was defined to handle VM localization and attain the optimal mapping between VMs and PMs in the search space. Finally, the sub-algorithms were combined on host overloading direction, VM choice and under-loaded host identification for VMs consolidation. But, it does not focus on reduce the VM migrations and maximize the resource use during VM consolidation.

Gholipour et al. [17] developed a novel data center resource handling process depending on the multiple conditions-based decision-making technique. Initially, the Joint VM and Container Multi-criteria Migration Decision (JVCMMD) strategy was designed. Then, a novel structure was constructed for controlling the JVCMM challenge. Also, multiple condition-based migration choice strategy was proposed to choose VMs to be shifted and lessen the power use, more VM migrations. On the contrary, it needs to consider other criteria such as memory, network bandwidth and storage size for increasing the VM consolidation efficiency.

Yun et al. [18] developed an adaptive harmony search algorithm for reducing the energy use of the data centre when enabling the stable process by means of VM consolidation. This algorithm aims a virtualized setting which allocates PM resources to a huge amount of VMs and accounts CPU, memory and network as the resources. Also, a migration cost was determined to minimize the energy use and prevent VM migration for stable tasks. On the other hand, it does not estimate the workload to maintain the unique patterns of VMs when increasing the efficiency of the system

3. Proposed methodology

This part describes the OH-BAC-FUP-MTR-MSCG in detail. This strategy minimizes the number of active PMs through consolidating the VMs to the smaller number of PMs for lessening the energy usage. The PMs in the cloud servers are of larger resource competence and assigning these larger resource larger resource competences to lower VM resource requests tends to inaccurate use of resources and also larger energy usage through operating high amount of PMs if a new request arrives. To solve this problem, the PM resources are partitioned into different classes with different resource competences for matching various VM requests. Also, dynamic VM consolidation scheme is used to consider the minimum active PMs by properly migrating VMs and decrease the resource utilizations. The description about this VM consolidation with OH-BAC-FUP-MTR method is given below. The block diagram of OH-BAC-FUP-MTR-MSCG method is portrayed in Figure 1.

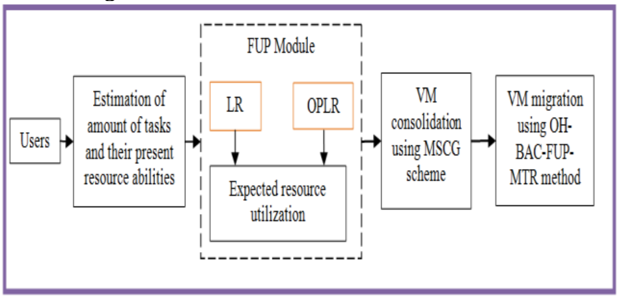

Figure.1 Block diagram of OH-BAC-FUP-MTR-MSCG method in heterogeneous cloud platform

3.1 System model

For a data center of heterogeneous PMs and VMs, the network's operating condition information is forecasted by the cloud server's system software. After, each data is forwarded to the cloud server. Based on these data, the system can update the parameters, schedule the VMs and handle the PM's state like assigning the new PM or shutting down an idle PM. The energy usage of a PM $\left(EC(u)\right)$ is assigned by its resource use u according to Eq. (1) where EC_{max} is the power used by an entirely-loaded PM & α is the fraction of inactive period of a PM.

$$EC(u) = \alpha EC_{max} + u(1-\alpha)EC_{max} \quad (1)$$

Observe that the use of a CPU, memory, storage size and bandwidth are time-varying. Also, it depends on the workloads on it. So, $u(t)$ is used and the overall energy used is determined as:

$$\xi = \int_{t_0}^{t_0+T} EC\left(u(t)\right)dt \quad (2)$$

In Eq. (2), t_0 is the initial time and T is the time during which a PM is active. It is considered that a cloud server has m categories of heterogeneous components, t_s refers to the interval that the VM consolidation initiates & t_e denotes the interval that VM consolidation terminates, f_k indicates the power used by the PM of category k per a given period that is in optimistic correlation with the traffic on it, called wl_k.

$$f_k \propto w_k \quad (3)$$

Consider $\mathrm{b_k}$ and $\mathrm{a_k}$ are the energy used by each machine of category k per unit interval before and after consolidation, respectively. They are determined as:

$\mathrm{b_k} = \mathrm{n_k} \times \int_{\mathrm{t_s-T}}^{\mathrm{t_s}} \mathrm{f_k}$

$\mathrm{a_k} = \mathrm{n_k} \times \int_{\mathrm{t_e}}^{\mathrm{t_e+T}} \mathrm{f_k}$ (4)

In Eq. (4), $\mathrm{n_k}$ is the number of machines of the $\mathrm{k^{th}}$ category. After, the energy used by VM migrations in a consolidation task is considered. This proposed strategy is encouraged by the factor of migration-rate defined as:

$\mathrm{h} = \mathrm{w_v} \times \int_{\mathrm{t_s}}^{\mathrm{t_e}} \mathrm{f_s dt} + \mathrm{w_v} \times \int_{\mathrm{t_s}}^{\mathrm{t_e}} \mathrm{f_d dt} + \mathrm{q}$ (5)

In Eq. (5), $\mathrm{w_v}$ denotes traffic of the migrated VMs, $\mathrm{f_s}$ and $\mathrm{f_d}$ are the power used by origin PM and target PM per a given period in VM migration and q refers to the highest power usage due to switching on a PM. If it does not require switching on a fresh PM as a target PM while a VM is migrated, then $\mathrm{q} = 0$.

According to this considerations & settings, the dilemma is modeled as:

$$max\, Er = \int_{t_s}^{t_e} \sum_{k=1}^{m} (b_k - a_k) - h$$

Subject to $\sum_{j=1}^{m} d_{ij} = 1, j = 1,2, \dots, u_j > 0$ (6)

In Eq. (6), d_{ij} is a Boolean variable for deciding whether i^{th} VM is localized on j^{th} PM. If i^{th} VM is localized on j^{th} PM, then consider $d_{ij} = 1$; or else, $d_{ij} = 0$ and u_j denotes the use of PM_j, PM_j should not be an empty PM. Also, Er is the energy conserved by the VM consolidation strategy i.e., power preserved by the VM consolidation by means of the restrains that each VM is localized on single PM and there are no inactive PMs.

3.2 Coalitional game-theoretic approach

A coalitional game Υ has a group of players $N = \{1,2, \dots\}$ and a quality q which characterises the rate generated by various subgroups of the gamers i.e., the reward of a coalition C. At this point, increasing the reward $q(C)$ defines increasing the power-gains of C.

$\Upsilon = (N, q)$ (7)

Gamers of the competition select to link or not to link C through choosing whether high power-gains is realized. To manage the Coalitional Game (CG) over C of PMs, PMs are split as 3 sets: S, E and L which involve PMs with superfluous-, extremely- and lowly-loaded, accordingly, based on different workload thresholds as:

$t_1 = Q_1$ and $t_2 = Q_3$ (8)

In Eq. (8), $t_1 = Q_1$ indicates the primary quartile of the workloads localized on each PM and $t_2 = Q_3$ indicates the 3rd quartile of the traffic localized on each PM. In this strategy, the fuse-and-partition-based CG is conducted for increasing q of C i.e., reward as:

$$max\, q$$

$$q = \frac{1}{n} \sum_{j=1}^{n} u_j$$

Subject to $0 < u_j \leq x_j, PM_j \notin S, PM_j \in C$ (9)

The use of C is defined as q which equivalents the mean use of PMs in C excluding the PMs having superfluous traffic. In Eq. (9), u_j is the real-time use of PM_j, x_j is the highest use allowed of PM_j and n refers to the amount of PMs in C excluding the PMs having superfluous traffic. In CG, fuse function defines the merging many PMs into a one C whereas the partition function operates in the opposed way in which traffic from a superfluous-loaded PM is shared via many PMs.

Eqns. (10)-(13) indicate the prerequisite for merging a superfluous- & a lowly-loaded PM, the partition of a superfluous-loaded PM, the merging of lowly-loaded PMs and the merging of PMs with heavy-loaded, accordingly.

$\forall PM_j \in S, PM_i \in L, C = \{PM_i, PM_j\}$

$q(C) > mean(u_j, u_i)$ (10)

$\forall PM_j \in S, u_j < q(C)$

$C = \{PM_i, PM_k\}, PM_i, PM_k \in L/S$ (11)

$\forall PM_j, PM_i \in L, C = \{PM_i, PM_j\}$

$q(C) > mean(u_j, u_i)$ (12)

$\forall PM_j, PM_i \in E, C = \{PM_i, PM_j\}$

$q(C) > mean(u_j, u_i)$ (13)

Where u_i is the use of PM_i. Observe that the functions enabled by the (10)-(13) prerequisites occur with the alphabetic manner of such prerequisites for guaranteeing that PMs having superfluous or lowly-loaded are controlled before those with the high workload. So, the objective of coalitional game is to create a PM set Υ which includes PMs that are operating in a high-efficiency state to conserve more energy.

$$\Upsilon = \{\mathrm{P}M_j | PM_j \in S \wedge u_j <= x_j\} \quad (14)$$

Algorithm:
Input: S, E, L
Output: Updated S, E, L
Begin
$\boldsymbol{for}(every\ PM_j\ in\ S)$
$\boldsymbol{for}(every\ PM_i\ in\ L)$
$\boldsymbol{if}(PM_i, PM_j\ are\ combined\ according\ to\ (10))$
Perform OH-BAC-FUP-MTR;
Migrate the source VM from PM_j to PM_i;
$\boldsymbol{end\ if}$
$\boldsymbol{end\ for}$
$\boldsymbol{end\ for}$
$\boldsymbol{for}(every\ PM_j\ in\ S)$
$\boldsymbol{if}(PM_j\ is\ partitioned\ according\ to\ (11))$
Perform OH-BAC-FUP-MTR;
Migrate the origin VM from PM_j to a fresh PM;
$\boldsymbol{end\ if}$
$\boldsymbol{end\ for}$
$\boldsymbol{for}(every\ PM_j, PM_i\ in\ L)$
$\boldsymbol{if}(PM_j, PM_i\ are\ combined\ according\ to\ (12))$
Perform OH-BAC-FUP-MTR;
Migrate the source VM from PM_j to PM_i;
$\boldsymbol{if}(PM_j\ is\ empty)$
Shut down PM_j
$\boldsymbol{end\ if}$
$\boldsymbol{end\ if}$
$\boldsymbol{end\ for}$
$\boldsymbol{for}(every\ PM_j, PM_i\ in\ E)$
$\boldsymbol{if}(PM_j, PM_i\ are\ combined\ according\ to\ (13))$
Perform OH-BAC-FUP-MTR;
Migrate the source VM from PM_j to PM_i;
$\boldsymbol{if}(PM_j\ is\ empty)$
Shut down PM_j
$\boldsymbol{end\ if}$
$\boldsymbol{end\ if}$
$\boldsymbol{end\ for}$

The above algorithm is used to create the coalition game. Each step creates coalitions so that PMs engaged are highly used. On the other hand, few PMs are shut down to conserve energy. Initially, PMs in S, L are decided and then PMs in E are chosen.

Workload fairness is an essential metric for analyzing the VM consolidation method that denotes the network's resource use. Assume wf as a measure of workload fairness.

$$wf = {(n_S + n_L)}/{n_E} \quad (15)$$

In Eq. (15), n_S, n_L and n_E are the number of PMs in S, L and E, accordingly. Based on Eq. (15), a lower wf denotes better workload fairness. Table 1 lists all the notations defined in this paper.

Table 1. List of notations

Symbols	Description
$EC(u)$	Energy usage of a PM assigned by its resource use u
EC_{max}	Power used by an entirely-loaded PM

α	Fraction of idle time of a PM
ξ	Overall energy used
t_0	Initial time
T	Time during which a PM is active
m	Types of heterogeneous machines
t_s	Interval that the VM consolidation initiates
t_e	Interval that VM consolidation terminates
f_k	Energy used by a PM of category k per unit period
wl_k	Workload of PM category k
b_k	Energy used by each machine of category k per unit interval before consolidation
a_k	Energy used by each machine of category k per unit interval after consolidation
n_k	Number of machines of the k^{th} category
h	Function of migration-cost
w_v	Workload of the migrated VMs
f_s	Energy used by source PM per unit time during VM migration
f_d	Energy used by target PM per unit time during VM migration
q	Highest energy usage as a result of turning on a PM
d_{ij}	Boolean variable for deciding whether i^{th} VM is localized on j^{th} PM
u_i	Use of PM_i
u_j	Real-time use of PM_j
Er	Energy conserved by the VM consolidation strategy
Υ	Coalitional game
N	Number of players
C	Payoff of a coalition
S	Superfluous-loaded PMs
E	Extremely-loaded PMs
L	Lowly-loaded PMs
t_1, t_2	Workload thresholds
Q_1, Q_3	Primary and third quartile of the workloads localized on every PM
x_j	The highest use allowed of PM_j
n	Number of PMs in the coalition excluding the PMs with superfluous load
wf	Workload fairness
n_S, n_L, n_E	Number of PMs in S, L and E, accordingly

4. Results and discussions

This part implements the OH-BAC-FUP-MTR-MSCG in CloudSim API 3.0.3. Also, its efficiency is compared with the existing methods such as CAM [15], EQ-VMC [16], JVCMMD [17] and OH-BAC-FUP-MTR [10]. The implementation scenario and their factors are given in Table 2.

Table 2. Implementation details

Type	Factor	Value
Host	No. of hosts	100
	Types of hosts	HP ProLiant ML110 G4
		HP ProLiant ML110 G5
HP ProLiant ML110 G4	No. of Processing Elements (PEs) per host	4
	Bandwidth	3Gbps
	Host memory	8GB
	Million Instructions Per Second (MIPS) of PE	2060
HP ProLiant ML110 G5	No. of PEs per host	4
	Bandwidth	3Gbps
	Host memory	8GB
	MIPS of PE	3560
VM	No. of VMs	450

	Types of VMs	High-CPU Medium Instance
		Extra Large Instance
		Small Instance
		Micro Instance
High-CPU Medium Instance	MIPS of PE	2500
	No. of PEs per VM	5
	VM memory	1GB
	Bandwidth	118Mbps
Extra Large Instance	MIPS of PE	2000
	No. of PEs per VM	4
	VM memory	4GB
	Bandwidth	118Mbps
Small Instance	MIPS of PE	1000
	No. of PEs per VM	3
	VM memory	2GB
	Bandwidth	118Mbps
Micro Instance	MIPS of PE	500
	No. of PEs per VM	2
	VM memory	1.5GB
	Bandwidth	118Mbps
Cloudlets	No. of tasks	500
	Length of task (Million Instructions (MI))	2500*simulation bound
	No. of PEs per demand	2
OH-BAC-FUP	No. of iterations	100
	No. of ants	5
	No. of honeybees	15
	α	0.8
	β	0.32
	γ	0.8
	ρ	0.1
	c	0.8

4.1 Energy consumption

It defines the total energy consumed by PMs at a unit time.

$$E_{total} = \int_t (x \times EC_{full} + (1 - x) \times EC_{full} \times u_i) \quad (16)$$

In Eq. (16), x denotes the % of energy used by the idle PMs, EC_{full} denotes the energy usage of entirely-loaded PM and u_i denotes the CPU usage of the PM.

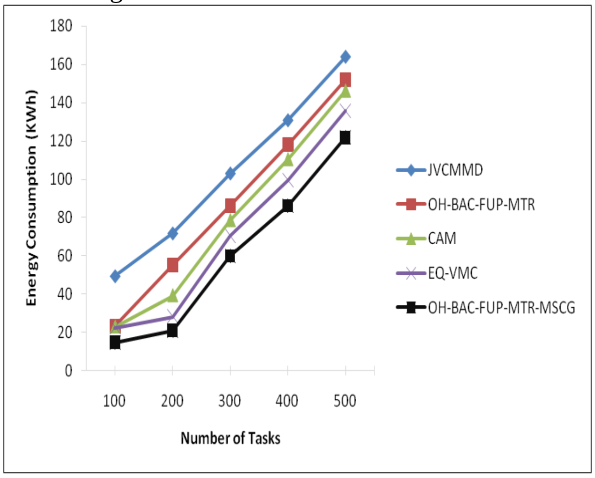

Figure.2 EC vs. No. of tasks

Figure 2 depicts the EC (in KWh) of different strategies with different number of tasks. This analysis indicates the OH-BAC-FUP-MTR-MSCG reduces the EC than the other strategies for VM consolidation and migration. For instance, energy consumed by PMs during 500 tasks for OH-BAC-FUP-MTR-MSCG is 25.57% less than the JVCMMD, 19.8% less than the OH-BAC-FUP-MTR, 16.59% less than the CAM and 10.12% less than the EQ-VMC strategies.

4.2 SLATAH

It is the fraction of period in which the active host utilizes 100% CPU.

$$\mathrm{SLATAH} = \frac{1}{n}\sum_{j=1}^{p}\frac{T_{s_j}}{T_{as_j}} \qquad (17)$$

In Eq. (17), p is the amount of PMs, T_{s_j} refers to the period in which j^{th} PM utilizes entire CPU & T_{as_j} refers to the overall number of j^{th} PM that is in the working state.

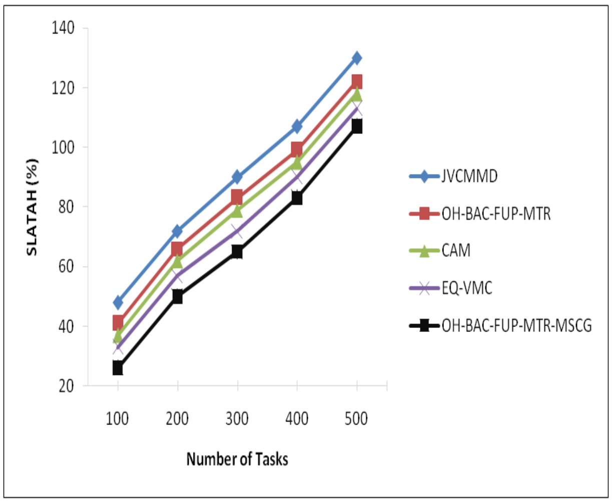

Figure.3 SLATAH vs. No. of tasks

Figure 3 illustrates the SLATAH (in %) of different strategies with different number of tasks. This analysis indicates the OH-BAC-FUP-MTR-MSCG reduces the SLATAH than all existing strategies for consolidating and migrating the VMs. For instance, SLATAH of OH-BAC-FUP-MTR-MSCG for 500 tasks is 17.7% less than the JVCMMD, 12.3% less than the OH-BAC-FUP-MTR, 9.3% less than the CAM and 5.3% less than the EQ-VMC strategies.

4.3 PDM

It is the total efficiency degradation because of migrating VMs.

$$\mathrm{PDM} = \frac{1}{m}\sum_{i=1}^{v}\frac{C_{d_i}}{C_{r_i}} \qquad (18)$$

In Eq. (18), v denotes the amount of VMs, C_{d_i} refers to the measure of efficiency deprivation of i^{th} VM generated by migrations and C_{r_i} indicates the entire CPU needed by i^{th} VM. Assume C_{d_i} is 10% of the CPU operation in MIPS agreed on SLA in all migrations of the i^{th} VM.

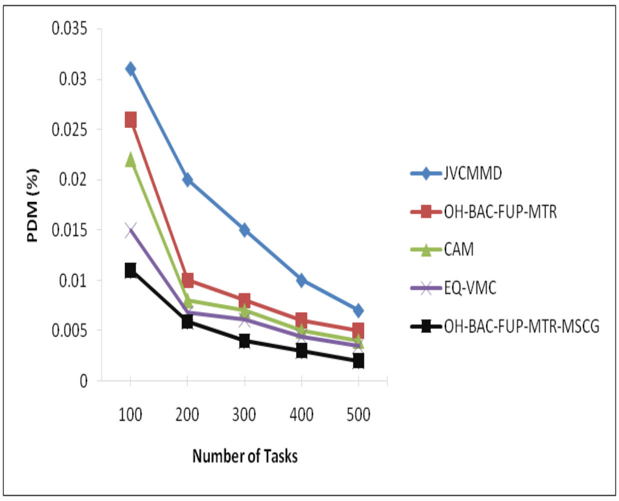

Figure.4 PDM vs. No. of tasks

Figure 4 depicts the PDM (in %) of different strategies with different number of tasks. This analysis indicates the OH-BAC-FUP-MTR-MSCG reduces the PDM than the other strategies for consolidating and migrating the VMs. For instance, PDM of OH-BAC-FUP-MTR-MSCG for 500 tasks is 71.43% less than the JVCMMD, 60% less than the OH-BAC-FUP-MTR, 50% less than the CAM and 42.86% less than the EQ-VMC strategies.

4.4 SLAV

It is applied to estimate the SLA distributed by the VM in the IaaS data center.

$$SLAV = SLATAH \times PDM \quad (19)$$

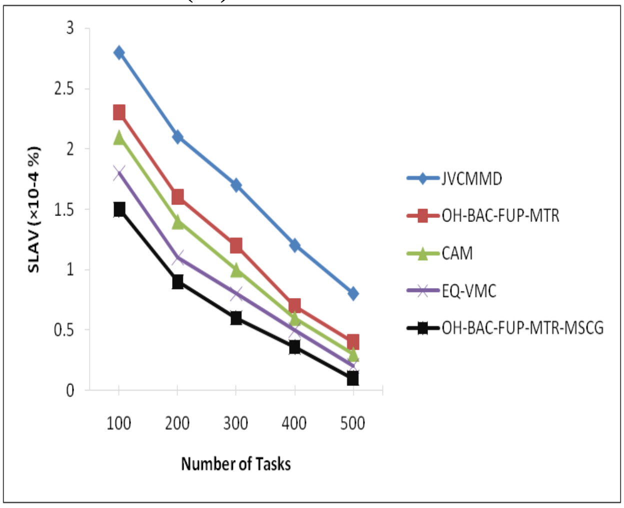

Figure.5 SLAV vs. No. of tasks

Figure 5 depicts the SLAV ($\times 10^4$ %) of different strategies with different number of tasks. This analysis indicates the OH-BAC-FUP-MTR-MSCG reduces the SLAV than the other strategies for consolidating and migrating the VMs. For instance, SLAV of OH-BAC-FUP-MTR-MSCG for 500 tasks is 87.5% less than the JVCMMD, 75% less than the OH-BAC-FUP-MTR, 66.67% less than the CAM and 50% less than the EQ-VMC strategies.

4.5 Number of VM migrations

It defines the amount of migrations generated in the remapping step.

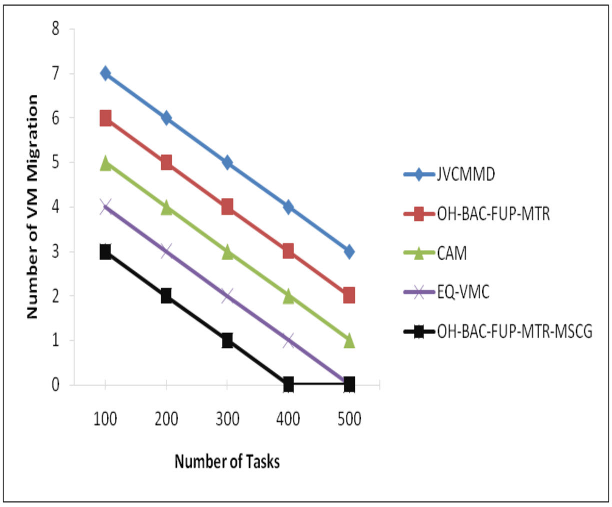

Figure.6 No. of VM migrations vs. No. of tasks

Figure 6 displays the No. of VM migrations for different strategies with different number of tasks. This analysis indicates the OH-BAC-FUP-MTR-MSCG reduces the No. of VM migrations compared to all existing strategies to consolidate and migrate the VMs. There are no VM migrations in OH-BAC-FUP-MTR-MSCG and EQ-VMC during 500 tasks whereas JVCMMD has 3 VM migrations, OH-BAC-FUP-MTR has 2 VM migrations and CAM has 1 VM migration.

4.6 Number of host's shutdowns

It chooses whether the hosts are turning ON or OFF. The host is turned OFF once the VMs are migrated. If each VM in a specific host is migrated, then the host is shutdown for minimizing the energy utilization. However, the host is becoming active while a VM has migrated to it again.

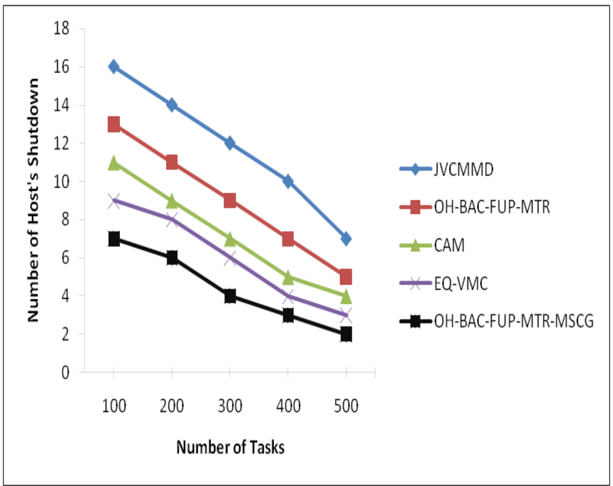

Figure.7 No. of host's shutdown vs. No. of tasks

Figure 7 portrays the No. of host's switched OFF for different strategies with different number of tasks. This analysis indicates the OH-BAC-FUP-MTR-MSCG reduces the amount of host's shutdowns compared to all existing strategies to consolidate and migrate the VMs. The amount of turned OFF hosts does not associates with the amount of hosts in the system. The host is turned OFF while it does not have any tasks after VM migrations. It indicates that the amount of host's turned OFF for OH-BAC-FUP-MTR-MSCG during 500 tasks is 2 whereas JVCMMD has 7 hosts' shutdowns, OH-BAC-FUP-MTR has 5 hosts' shutdowns, CAM has 4 hosts' shutdowns and EQ-VMC has 3 hosts' shutdowns. It means if a host is shut down, then it continues this stage for a while; therefore the effectiveness of OH-BAC-FUP-MTR is maximized.

5. Conclusion

This paper proposes a VM consolidation strategy along with OH-BAC-FUP-MTR to reduce the EC of PMs effectively. This strategy migrates the VMs to the smaller number of PMs by consolidating them based on the workload and resource utilization to lessen the power use and promote green computing. First, the PMs are partitioned into different sets according to their workload levels and then a new fuse-and-partition-based coalitional player strategy is employed for deciding associates from these sets to generate effective coalitions. Then, OH-BAC-FUP-MTR is conducted amongst the coalition associates to increase each coalition's reward and PMs are handled to functioning in the maximum power-efficient condition. To conclude, the findings proved that the proposed strategy accomplishes an overall EC of 60.63Kwh, SLATAH of 66.2%, PDM of 0.0052%, SLAV of 0.692×10^{-4}%, 1 VM migration and 4 host's shutdowns than the other VM migration strategies. However, the load-balancing is affected by the constant fluctuations of VM resource demands. So, the future work will focus on handling the fluctuations of the resource demands to enhance the efficiency of load-balancing in cloud computing.

Times is specified, Times Roman of Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts if possible. True-Type 1 fonts are preferred.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

This paper investigation, resource, data Corrections-writing original data preparation, writing review and editing, implementation has been done by 1st author. The conceptualization, methodology, software validation formal analysis has been done by 3nd author. The supervision and research guide and administration have been done by 2nd author.

Acknowledgement

we are grateful thanks to Chikkanna Government Arts college tirupur, Pg and Research department computer science for providing a favorable environment to carry out of our research work.

References

[1] Y. Zhou, D. Zhang, and N. Xiong, "Post-cloud computing paradigms: a survey and comparison," Tsinghua Science and Technology, Vol. 22, No. 6, pp. 714-732, 2017.
[2] E. J. Ghomi, A. M. Rahmani, and N. N. Qader, "Load-balancing algorithms in cloud computing: a survey," Journal of Network and Computer Applications, Vol. 88, pp. 50-71, 2017.
[3] M. A. Shahid, N. Islam, M. M. Alam, M. M. Su'ud, and S. Musa, "A comprehensive study of load balancing approaches in the cloud computing environment and a novel fault tolerance approach," IEEE Access, Vol. 8, pp. 130500-130526, 2020.
[4] B. Alankar, G. Sharma, H. Kaur, R. Valverde, and V. Chang, "Experimental setup for investigating the efficient load balancing algorithms on virtual cloud," Sensors, Vol. 20, No. 24, pp. 1-26, 2020.
[5] M. Villari, M. Fazio, S. Dustdar, O. Rana, and R. Ranjan, "Osmotic computing: a new paradigm for edge/cloud integration," IEEE Cloud Computing, Vol. 3, No. 6, pp. 76-83, 2016.
[6] M. Villari, A. Celesti, and M. Fazio, "Towards osmotic computing: Looking at basic principles and technologies," In: Conference on Complex, Intelligent, and Software Intensive Systems, Springer, Cham, pp. 906-915, 2017.
[7] M. Gamal, R. Rizk, H. Mahdi, and B. E. Elnaghi, "Osmotic bio-inspired load balancing algorithm in cloud computing," IEEE Access, Vol. 7, pp. 42735-42744, 2019.
[8] R.GowriPrakash, R.Shankar, and S.Duraisamy, "FUPA: future utilization prediction algorithm based load balancing scheme for optimal VM migration in cloud computing," In: IEEE Fourth International Conference on Inventive Systems and Control, pp. 638-644, 2020.
[9] R. Cziva, S. Jouët, D. Stapleton, F. P. Tso, and D. P. Pezaros, "SDN-based virtual machine management for cloud data centers," IEEE Transactions on Network and Service Management, Vol. 13, No. 2, pp. 212-225, 2016.
[10] R.GowriPrakash, R.Shankar, and S. Duraisamy, "Resource utilization prediction with multipath traffic routing for congestion-aware VM migration in cloud computing," Indian Journal of Science and Technology, Vol. 14, No. 7, pp. 636-651, 2021.
[11] Y. Liu, B. Gong, C. Xing, and Y. Jian, "A virtual machine migration strategy based on time series workload prediction using cloud model," Mathematical Problems in Engineering, Vol. 2014, pp. 1-11, 2014.

[12] V. De Maio, G. Kecskemeti, and R. Prodan, "A workload-aware energy model for virtual machine migration," In: IEEE International Conference on Cluster Computing, pp. 274-283, 2015.
[13] P. Arroba, J. M. Moya, J. L. Ayala, and R. R. Buyya, "Dynamic voltage and frequency scaling-aware dynamic consolidation of virtual machines for energy efficient cloud data centers," Concurrency and Computation: Practice and Experience, Vol. 29, No. 10, pp. 1-13, 2017.
[14] L. Guo, G. Hu, Y. Dong, Y. Luo, and Y. Zhu, "A game based consolidation method of virtual machines in cloud data centers with energy and load constraints," IEEE Access, Vol. 6, pp. 4664-4676, 2017.
[15] W. Wu, W. Lin, and Z. Peng, "An intelligent power consumption model for virtual machines under CPU-intensive workload in cloud environment," Soft Computing, Vol. 21, No. 19, pp. 5755-5764, 2017.
[16] Z. Li, X. Yu, L. Yu, S. Guo, and V. Chang, "Energy-efficient and quality-aware VM consolidation method," Future Generation Computer Systems, Vol. 102, pp. 789-809, 2020.
[17] N. Gholipour, E. Arianyan, and R. Buyya, "A novel energy-aware resource management technique using joint VM and container consolidation approach for green computing in cloud data centers," Simulation Modelling Practice and Theory, Vol. 104, pp. 1-17, 2020.
[18] H. Y. Yun, S. H. Jin, and K. S. Kim, "Workload stability-aware virtual machine consolidation using adaptive harmony search in cloud datacenters," Applied Sciences, Vol. 11, No. 2, pp. 1-23, 2021