International Journal of Distributed and Cloud Computing
7 (2) December 2019, 07-18
http://www.publishingindia.com/ijdcc

Task Scheduling and VM Allocation in Cloud Computing: A Survey

Jeny Varghese[1*] and S. Jagannatha[2]
[1]Research Scholar, Department of MCA, MS Ramaiah Institute of Technology, Affiliated to VTU, Belagavi and Assistant Professor, PES University, Bangalore, Karnataka, India. Email: jens4u@gmail.com
[2]Associate Professor, Department of MCA, MS Ramaiah Institute of Technology, Affiliated to VTU, Belagavi, Bangalore, Karnataka, India.
*Corresponding Author

Abstract: Cloud computing is a fast-emerging area with so many benefits for the end users. Many leading firms make use of cloud mainly to store their huge volume of data and other such purposes. When speaking about cloud, we need to focus on the data centers. This is because the virtual machines (VM) present in the cloud are mapped to the physical devices that are present in the data centers. This seems to be a simple process, but the tricky part is when the number of users increases. Number of users send the request for a particular or more than one service to the cloud. The service providers have to handle the entire request and also concentrate on the satisfaction of the end user along with the quality of service (QoS). Most important thing is that the service provider must give a quick response to the cloud consumers, i.e., the delay must be minimized. Only then the QoS provided could be maintained. In this paper, we are going to see about how the tasks are being scheduled and allocated optimally to the VMs present in the cloud. We are going to delineate the various algorithms that are implemented so far for this process. In this paper, we have described an overview of cloud, task scheduling and VM allocation.

Keywords: Cloud computing, Data center, Service providers, Task scheduling, VM allocation.

I. Introduction

Cloud computing is a recent, growing technology with the ability to produce lot of resources as utility services. People pay for hardware and software as per their use. Similarly in cloud computing, services are given according to the user request [1]. Normally, cloud computing is mostly utilized for sharing the storage, networks and services. This area has certain special characteristics like service anyplace, enormous resources and many others. The cloud users must ensure the quality of the resources givesn by the service providers. The three main and significant services of cloud include: *Infrastructure-as-a-service* (Iaas), *Platform-as-a-service* (Paas) and *Software-as-a-service* (Saas) [2].

Cloud computing is deployed in three different categories as: *private*, *public* and *hybrid*. In order to achieve the full range of benefits, these three categories exist [3]. The *private* cloud can also be called as enterprise cloud as it can be accessed only by the organization. These clouds are exclusively utilized by the owner and are present in their premises. This model provides a higher degree of security and controls over the resources stored in the cloud. The *public* cloud can be accessed publicly and supervised by third-party service providers. Since, it is available, open security cannot be guaranteed to a great extent. The *hybrid* cloud, as the name suggests is the combination of the above models. It allows the user to store confidential data and facts in private cloud and the rest in public cloud. The only problem of this model lies in the classification of data as which is more important and the reverse. The *community* cloud involves the distribution of computing resources among the organizations of the same community. The main advantage of this model is that it reduces the costs for its setup as it is distributed among the organizations and also has high security [4]. These deployment models and the services offered by the cloud are shown in Fig. 1.

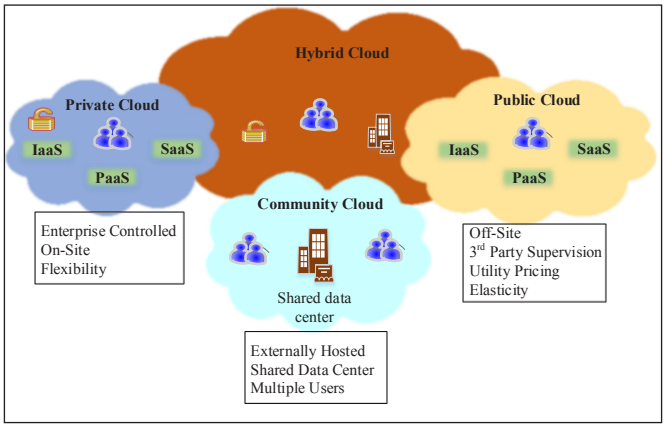

Fig. 1: Cloud Deployment Model and Services

A. Definitions

In the following, we have provided the definitions of the entities present in the cloud computing environment.

- *Definition 1: Physical Machine (PM)*

  PMs are those present in the data centers that consist of more than one VM to complete the user task. Depending on the workload, these VM are allocated to the PM to respond to the application requests.

- *Definition 2: Virtual Machine (VM)*

  PMs do the task assigned by the user at any cost. VM assists the PM in terms of computing resources as server, storage, etc. VM process is separated into two types as Hardware and Application virtualization. Multiple VM can be started and stopped on a single PM.

- *Definition 3: Cloud Service Provider*

  The cloud service providers are generally organizations or people that provide distributed sources for the consumers who have subscribed for that service. In common, the services are provided as either on-demand or pay-per-use system. The consumers can access these services from the cloud through the internet.

- *Definition 4: Cloud Consumers*

  The people or organizations that consume or make use of the services from the cloud are the users. This category of cloud computing entity browses the service from the service provider, set up Service Level Agreement (SLA) and then uses that service. The payment is made as per the use accordingly.

- *Definition 5: Virtualization*

  It is the technology of logically separating the PM of a server into several isolated machines as VM. Virtualization allows creating an abstract layer of system resources as well as enhances the hardware independence. In general, there are four types of virtualization as: Desktop, Application, Network and Storage.

B. Problem Analysis

In recent days, task scheduling and VM allocation are becoming the leading aspects in the area of cloud computing. Choosing the best algorithm with appropriate parameters still remains a huge challenge in the research community. Task scheduling is the process of assigning the tasks to the appropriate resources present in the cloud. For instance, consider there are "n" number of tasks arriving from the "s" set of users for "y" physical resources, then the "F" objective function for scheduling involved in (Eq. 1) [5].

$$F = \sum_{k=1}^{n} X_k * Y_k \ \forall U_k \quad (n = x \text{ or } x) \tag{1}$$

Here, F represents the objective function for task scheduling, X represents the tasks, Y represents the resources and U represents the user.

VM Allocation is the process of mapping the tasks with the respective resources. For instance, consider $Y = \{y_1, y_2, y_3 \ldots y_n\}$ be the resources of the VMs and $X = \{x_1, x_2, x_3 \ldots x_n\}$ be the tasks that needs the resources.

$$\forall y \in Y \rightarrow \{\text{memory, bandwidth, CPU}\} \tag{2}$$

Thus, the objective function for VM is given in (Eq. 3)

$$S = \sum_{k=1}^{n} Y_k \rightarrow X_k \ \forall k \left(n = x \text{ or} \neq x\right) \tag{3}$$

Here, S represents the objective function for VM.

In general, cloud consumers make use of more than one VM for each and every task they request from the cloud. The tasks are scheduled solely for the purpose of reducing the delay and increasing the performance of cloud. If a scheduler is considered to be effective, then it must adjust its scheduling technique under any altering conditions and also to any kind of cloud consumer's requirement [6]. Task scheduling is done into arrange the many tasks that arrive at a particular instant and decide the task that gets the response first. In general, task scheduling is done in the following steps:

- At first, user will send the request for a service in the cloud.
- The task is stored in a queue according to the arrival time.
- The task scheduler will segregate the tasks that will get the response by any of the algorithm.
- After that cloud will provide the resources requested by the task.
- The cloud will ensure quality as well as customer satisfaction.

Task scheduling is the most sought after research area in cloud computing. One possible solution for the problem of task scheduling is to change the number of tasks and datacenter automatically [7]. In this survey, we have given a detailed survey of various approaches used for scheduling the user requests so far. The main feature of our survey relies on the scheduling algorithms and the Meta heuristic algorithms. Before this survey, we have identified some leading challenges as well as the importance of this research study is given as follows:

- Reducing the energy consumed and increasing the resource utilization for a particular task.
- Tasks with unbalanced utilization of resources often leads to wastage of resources.
- The task requested by the user must be completed within the specified time. Services must be provided according to the task specification (i.e.,) tasks, which are critical or deadline sensitive have to be responded first by the service provider.
- Cloud computing itself holds a lot of issues like availability and allocation of physical resources such as storage, service, network and others.

VMs are connected to the physical devices and give provisions to run applications like a separate computer. Virtualization is one of the important technologies of cloud, which involves sharing of resources with the users. In certain cases where the resources are given in excess to the tasks that require only minimum amount of resources, it leads to the over provision

of resources. To avoid these problems, load rebalancing is introduced. Load balancing is nothing but mapping the resource requests from user to appropriate VMs in an optimal manner. This process improves resource utilization and reduces the time takes back to respond [8].

II. Literature Survey

A. Task Scheduling

The cloud consumers access the required resources virtually with the help of web. Many novel optimization algorithms are used for performing the task scheduling in cloud. Task scheduling still remains as the significant challenge in the field of cloud computing. This is nothing but managing the requests from the end user properly and satisfying their needs. The major issue in task scheduling is the efficient allocation of resources to the tasks requested by the user [9]. A short overview of all the algorithms presented in this paper is given in the flow chart mentioned in Fig. 2.

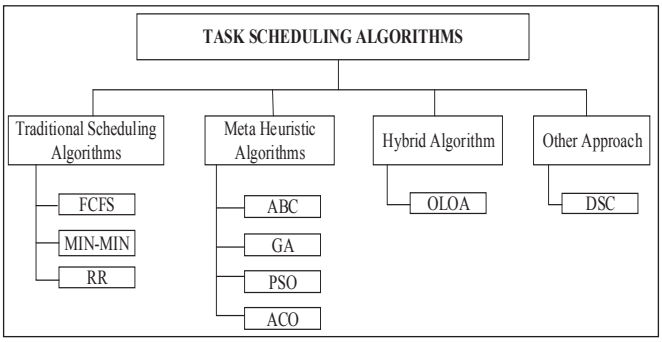

Fig. 2: Algorithms for Task Scheduling

A novel comparison of the leading algorithms used for task scheduling so far. Algorithms with single objective do not deliver effective results comparable to those with multiple objective functions. Also, efficient scheduling algorithm can yield proper response to the user's request and at the same time increase the performance of the cloud environment. Task scheduling is done mainly with the objective of reducing the execution time of the tasks and maximizing the resource utilization by the tasks. More satisfactory results can be achieved by adding two or more metrics into the algorithm. First Come First Serve (FCFS) is the fundamental technique that uses only the arrival time to schedule the tasks and doesn't consider any other parameter. Min-Min algorithm is a scheduling technique that works on the strategy that the task with a minimum execution time is selected for all tasks. If a very short task follows a lengthy task, then the task has to wait until the completion of lengthy task [10].

Artificial Bee Colony (ABC) algorithm is a heuristic method of finding and sharing the resources available in the cloud. The tasks are shared for the user's tasks. Initially, the workload of the server and wants of the users are identified through the previous resource utilization and size of the VM allocated. Based on this, the expected completion time for each server is calculated. Finally, the servers are classified based on deadline and resource requirement and then each task is assigned to suitable server. When users submit their request it is classified by the cloud service broker. The server is categorized by the ABC algorithm. Tasks with short deadline and high resource size is allocated to server with less workload when tasks with high priority arrives, the task with low priority are preempted. Though, it provides an efficient scheduling algorithm, it may not be efficient in certain cases due to the limited consideration of parameters [11]. A detailed view of task scheduling and how it is allocated in the VM is given in Fig. 3.

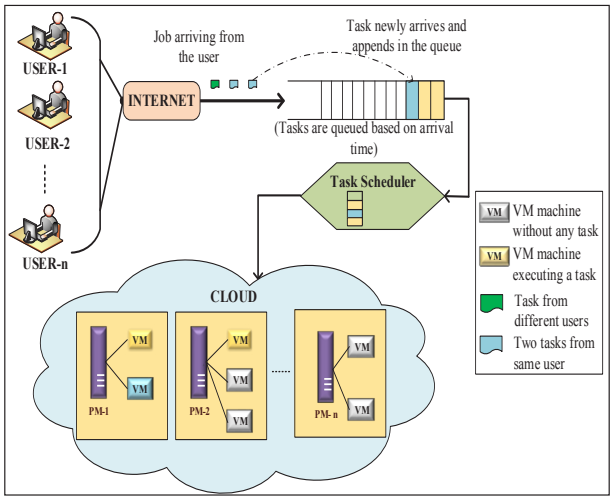

Fig. 3: Task Scheduling and VM Allocation Process

Task scheduling boosts the performance of the cloud and minimizes the cost for operation. As already seen, single algorithm cannot provide much efficient result. Thus, a hybrid algorithm of Lion Optimization Algorithm (LOA) and Opposition Based Learning (OBL) is combined to perform the operation of task scheduling. Here, the makespan and cost are utilized to optimize the tasks and resources. First, the scheduling is done in parallel and all the tasks are processed simultaneously by the cloud. The scheduling algorithm decides the number of resources required to complete a particular task. A larger task is spilt into sub tasks before sending to parallel processing. It has been shown that this algorithm has overwhelmed the early optimization algorithms and tends to show good improvement in performance. Though the results seem convincing, the arrival time and priority could be considered to make it more efficient [12].

The tremendous growth of cloud computing has paved the way for increased network traffic. Load balancing allocates the incoming traffic between different computers to do huge number of work in lesser time. It became a vital part which could be done either by scheduling the tasks or by migration the VMs according to the workload. In this paper, a multi objective Particle Swarm Optimization is put forward for scheduling the tasks. Swarm intelligence algorithms are very popular in solving NP hard problem. PSO is based on a stochastic technique. It is necessary that a system in the cloud environment must be error

free and tolerant to the various types of faults arising in the system. The main reason for choosing this algorithm is that it contains no evolution probabilities as in other Meta heuristic algorithms [13].

Standard genetic algorithm has some drawbacks with respect to population evolution and so an improved genetic algorithm is applied for scheduling the tasks. The improvement is done in the execution of genetic algorithm and making use of binary encoding for chromosomes. However, the determination of crossover and mutation remained tedious and not optimal probability. In genetic algorithm, when the population evolves to the middle and late stages of the algorithm, its diversity is destroyed. In this paper, a new fitness formula has been put forth that improves the early evolutionary population and doesn't allow it to occur in crossover and mutation process. After that, it becomes adaptive genetic algorithm with the advantage of automatically increase and reduce the probability as per the fitness value. Although fitness is changed, genetic algorithm is generally not good with the initial population and also has the difficulty with equality constraints [14].

Ant Colony Optimization (ACO) is an effective technique for solving the NP-complete problems and gets its inspiration from the behavior of ant colonies for searching food. A non-dominant sort approach is used to solve multi-objective functions. Two fitness functions are calculated namely makespan and load balancing. Here, certain assumptions are made before proceeding with task scheduling as no deadline constraint, estimated execution time, independent tasks, etc. This is done to obtain better level of load balancing. Starting from the initial stage, each task is assigned to a specific resource and pre-emption is not allowed. Ants construct solutions by moving from one VM to another until the tour is completed. This method is applicable only for the tasks that satisfy the premade assumptions. When a random task enters the cloud, this approach does not provide an optimal scheduling of resources for the task [15].

Task scheduling is done based on round-robin (RR) algorithm. First of all, the static quantum time is hard to predict as either long or short. If time slice is long, then RR becomes FCFS otherwise, delay in processing the user's request. Hence in this paper, time slice is calculated for each round instead of each task to reduce the complexity of the algorithm. Also, tasks are queued based on the size of their burst time and the median is chosen as the quantum time. While ordering the tasks, those tasks that are adjacent to the median are not placed in the next round. The burst time of the next task adjacent to the median is selected as the quantum time for each round of scheduling. Quantum time is dynamic and calculated as a median time. Time slice is calculated for each round to increase the performance than the previous and traditional round robin algorithms. This algorithm improves the scheduling results of the same priority tasks that enter the cloud. Even though time slice is given importance, many important parameters are not considered which may not provide optimal solution [16].

In this paper, author proposed a novel approach that uses Dominant Sequence Clustering (DSC) and a Weighted Least Connection (WLC) for scheduling the tasks. Initially, tasks are prioritized based on the constraints of deadline and makespan. After that, the tasks from the users are clustered. After clustering, the tasks are arranged in a graph and ranked using Modified Heterogeneous Earliest Finish Time (MHEFT) algorithm. More than one cluster can be present in the graph. Tasks are rescheduled and ranked to determine highest priority and scheduled first for the next process. Load balancing is also performed by the WLC algorithm to distribute the load considering the server weight, capacity and the client connectivity. The highest priority task is sent to the VM for processing. The server is selected as either highly weighted or lease connected to increase the response time. Resource utilization is increased with the process of clustering the tasks and also provide optimal scheduling [17].

Cloud computing is on the top of the research trends due to its high potential and flexible nature. Due to massive volume of user requests, scheduling the tasks that is profitable for service providers and the customers is difficult. This problem of scheduling is one of the most important and challenging issues of cloud computing. Cloud has a scheduler to monitor and distribute workloads as well as to adapt resources to the requests. A detailed review of some recently proposed algorithms in the field of task scheduling is described. Nearly 15 different approaches are dealt in detail for task scheduling. Each method has its own pros and cons. The main focus of the paper is to schedule the tasks by allocating the resources available in the cloud and the data centers. Tasks are scheduled through an appropriate framework for resource allocation [18].

The increase in the use of cloud and virtualization technologies has led to the large scale data centers. This paper describes reducing energy without compromising the deadline for tasks. A multi-objective function for task scheduling and VM allocation optimally is achieved in this paper. In addition to makespan and energy, it is aimed to achieve good load balancing and high utilization. Here, the tasks are prioritized dynamically and then scheduled to a suitable resource. The resources are selected using hybrid Meta heuristic algorithm. Initially, the tasks are stored in queue according to it waiting time and applied Dynamic Classifier Algorithm considering length, deadline, waiting time and burst time. Tasks with critical deadline are scheduled firstly and immediately. Thus, achieves zero deadline violation [19]. Parameters considered in various approaches are tabulated in Table I.

B. Comparison Analysis of Task Scheduling Algorithms

So far, we have seen about the conventional as well as Meta heuristic approaches for scheduling the tasks that arrive at the cloud. A detailed view of all the algorithms discussed here is presented in Table II. While using conventional schedulers, there are certain drawbacks that affect the cloud service in spite of the certain benefits. The general drawbacks in using the traditional algorithms are as follows:

TABLE I: EVALUATION PARAMETERS/METRICS OF ALL ALGORITHMS

Methods	Arrival Time	Task Size	Execution Time	Makespan	Cost	Resource Utilization	Deadline
FCFS [9]	Yes	No	No	No	No	No	No
RR [15]	No	No	Yes	No	No	No	No
Min-Min [9]	No	Yes	No	No	No	No	No
PSO [12]	No	No	No	Yes	Yes	No	Yes
GA [13]	No	No	Yes	No	No	No	No
ABC [10]	No	Yes	Yes	No	No	No	Yes
ACO [17]	No	No	No	Yes	No	Yes	No
OLOA [14]	No	No	No	Yes	Yes	Yes	No
DSC [16]	No	No	No	Yes	No	Yes	Yes

- While considering tasks, instead of dealing with a number of constraints, any one of the constraints is alone considered. Single parametric functions do not provide optimal results at some cases.
- Conventional schedulers did not consider the significant metrics for task scheduling. This results in large delay in order to complete the execution of all the tasks.
- With schedulers, efficient results are obtained when the arriving tasks are of same type and size than the random tasks. For instance, if tasks from multiple users point to same resource then the tasks are scheduled effectively.
- Conventional schedulers achieve optimal results to some extent but in most cases result in improper load balancing.

In order to overcome these problems, Meta heuristic approaches were followed. These algorithms definitely provide optimal results. The advantages of using Meta heuristic algorithm listed as follows:

- As seen already, multi-objective function yields better results than the opposite (single objective function). In these algorithms, more than one parameter can be considered.
- Meta heuristic algorithms provide different solutions for different problems and also boost up the result.
- Scheduling problems are NP-hard problems and Meta heuristic algorithm proved to provide optimal solution within acceptable time [20].

In Meta heuristic algorithm, search space is large. So, it is combined with other algorithm to yield better results. The Meta heuristic algorithm which gives the global optimum solution usually has high time complexity. Apart from this, several other approaches were used to effectively schedule the tasks that are exact to get the resources from the VM.

TABLE II: SUMMARY OF THE ALGORITHMS FOR TASK SCHEDULING

Algorithm	Description	Pros	Cons
Traditional Scheduling Algorithms			
Scheduling algorithms [1] FCFS, Min-Min	Uses only one of the parameters of task scheduling and performs the process.	• Simple scheduling technique. • Fast execution.	• Delay is increased. • Long tasks wait for long time.
MDRR	Selects the time slice based on median value of number of tasks.	• Less overhead. • High Performance.	• High Delay and turnaround time. • Limited parameter consideration.
Meta Heuristic/Optimization Algorithms			
GA	Performs task scheduling with the diversity of population and achieve good result with modified fitness function.	• Minimize the total task time. • Balance the load of computing resources.	• Lot of time and space as task increases. • Limited in operators.
PSO	Achieve load balancing by distributing the traffic among backend servers.	• No crossover and mutation evolution. • PSO is used as an optimizer.	• Does not handle scattered fitness function. • Too slow.

Algorithm	Description	Pros	Cons
ABC	Priority based task scheduling and work allocation for servers.	• High availability of resources. • Intelligent solution.	• Did not describe resource utilization. • Did not consider task deadline.
ACO	Tasks are scheduled in each iteration and a global optimum solution is obtained.	• High Resource Utilization and Load balancing. • Reduced Makespan.	• Limited parameters consideration. • Doesn't consider task deadline.
Hybrid Algorithm			
OLOA	Hybridization of lion optimization algorithm and opposition based learning algorithm.	• Reduce the cost of the operation. • Less execution time.	• Dynamic priority. • Doesn't consider resource size.
Other Approaches			
DSC	To construct graph, tasks are clustered by priority.	• Deadline constraint tasks are executed first. • Increase response time.	• High complexity in graph construction. • Less parameter for task ranking.

C. VM Allocation

Multiple VM can be assigned to a single PM, but it lead to the problem of overloading. Each and every VM could handle several application processes and provides enhanced service multiplexing and resource utilization. Energy consumption of data centers is greatly reduced by the use of virtualized resources [21].

With the virtualization technology, the cloud provides a lot of service to multi user environment. There are chances for the security attacks like side channel attacks. Instead of paying attention to eliminating the side channel attacks, the deployment of VM could be focused. To reduce the power consumption, a novel Previous Selected Server First (SC-PSSF) method is implemented for VM allocation as well. Generally, VM allocation is done in two stages as initial allocation and reallocation by considering the security. Here, the load balancing is done by spreading VM to the hosts with an average distribution. However, if the previous selected server is overloaded with tasks then this algorithm may not give efficient results [22]. A detailed survey of the entire algorithms presented in this paper is given in Fig. 4.

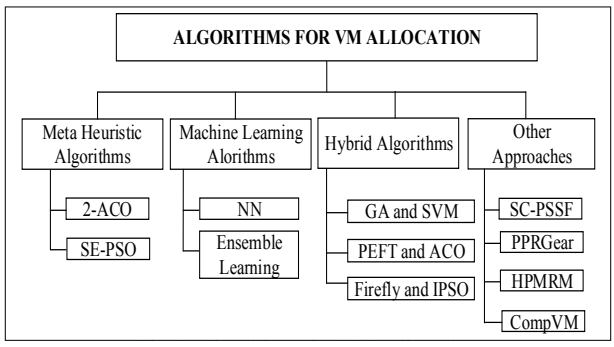

Fig. 4: Algorithms for VM Allocation

In order to optimize the VM allocation process, Genetic Algorithm is combined with Support Vector Machine. The availability of resource is checked by using the Modified Best Fit Decreasing (MBFD) algorithm. After that, the Genetic Algorithm is used to optimize the performance of MBFD and for the cross-validation Polynomial Support Vector Machine (P-SVM) is implied. Virtualization functions on the software layers and offers Virtual Machine Monitors (VMM) within the OS and hardware and also maps the machines on different platforms. The VMM performs the resource allocation. By using virtualization the utilization of resources accessible to the user can be improved. The high performance of data centers is achieved along with the energy consumption. The availability of resources and the allocation of PM are done by MBFD. This is optimized by the Genetic Algorithm to minimize the system overload and evaluating each machine to check the load. SVM performs the cross validation and it if true allocation is set else passed on to the GA. The increase in number of algorithm leads to time complexity and with GA the results may not be optimal all the time [23].

To achieve Hybrid approach Deadline-constrained, Dynamic VM Provisioning and Load Balancing (HDD-PLB), the hybridization of heuristic algorithm with Meta heuristic approach is followed. Two different heuristics are hybridized with the ACO. Load balancing techniques make sure that each data center performs equal number of tasks. Both the heurists, Heterogeneous Earliest Finish Time (HEFT) and Predict Earliest Finish Time (PEFT) are list-based scheduling approaches. First, the results of the PEFT are given to the ACO and VM allocation is performed. After that, the results of the HEFT are given to the ACO to perform the VM allocation. It is concluded that PEFT produces better results than HEFT in terms of load balancing optimization. But, it is not sure that PEFT can handle heterogeneous VM allocation effectively. When tasks arrive at

random or at large volume, the earliest finish time is quite hard to find [24].

Cloud environments encounter challenges with balancing the load on the server. A hybrid approach of combining the Firefly and Improved Particle Swarm Optimization (IPSO) is presented. This method is proposed to achieve efficient response accuracy. The firefly algorithm has a high convergence rate. The firefly algorithm will slow down near the optimal point of the search process. If the initial population is not reached, then convergence rate is not good. For this reason, the firefly algorithm is proposed for initializing the best initial population and for scheduling the tasks IPSO is used and selects the best execution order. Thus, the drawbacks of each algorithm are balanced by each other. The flexibility in load minimization is better. But, the algorithm implied are too slow in performance. Also, only the initial stage of the load balancing is mentioned here and the load rebalancing is not discussed which remains as a significant aspect [25].

A neural network-based adaptive selection of VM consolidation algorithms is proposed. Here, the algorithms are chosen appropriately according to the cloud provider's priority of goal and the environment parameters. The resource utilization and workload distribution imbalance are calculated from average and standard deviation CPU utilization of active hosts. The dataset generation is executed based on the normalized evaluation result of randomized evaluation priority. First, the raw dataset is generated for several timestamps and then full dataset is generated. Random evaluation priority is used in input layer and training and testing datasets are generated. Finally, the performance score is compared with original methods. However, there is no dedicated system for the dataset generation. Also, there is no proof that it adapts and provides the best results to the new improvised algorithms when given as input to the adapter [26].

Predictive Interference and Energy Aware (PIEA) ensemble-based consolidation algorithm is proposed. It uses a predictive modeling to classify the workloads on the server using resource usage features to make more informed consolidation decision. The ensemble learning is implied for this process that plays a pivotal role for improving predictive performance for many different problems in a multi-tenant cloud environment. In addition to Logistic Regression, they also consider three ensemble approaches as bagging, boosting and stacking. First, allocate the VM to the hosts whose interference effects are reduced and then the amount of PM are minimized in a way that improves the energy efficiency and delivery of service guarantees. The main objective of this paper is to classify the workload for reducing the energy consumed during the VM allocation. They have not described about the response time or resource utilization by the VM in the cloud environment [27].

With energy consumption growing as a leading concern in the area of cloud computing, a strategy to calculate the optimized working utilization levels for host computers is proposed. This approach considers the performance to power ratio as (PPR) for various host types. This method managed to achieve an optimal balance between the host utilization and energy consumption. With the highest value of PPRGear, the energy consumption is tremendously reduced. When a computing node has higher gear ratio than the preferred, it is over-utilized and one or multiple VM is selected and migrated out. On the other hand, if a node is having a lower gear value then it is underutilized, and now the cloud migrates out all the VM from that particular host and shut down that host completely. Though this algorithm focused on the energy and performance of the allocation process, it doesn't provide a hint about the cost of the operations [28].

A phase-wise optimization approach with two ant colony base approach is proposed for jointly optimizing the energy consumption of servers and network elements. While constructing the solution, the layout of the datacenter and the communication patterns of the application are considered. The data centered network is represented as a weighted graph, with servers and switches as vertices. The power consumption reduction is focused only on the physical servers present in the data centers. First, the VM present in the networks are optimally scheduled and then the communication flow is routed to the different tenants. The path is constructed for the flow by picking one node after another probabilistically. If the energy consumed by one solution is less than the energy consumed by the best solution, then it is updated as the best solution and the iteration continues [29].

VM migration in a federation of cloud environment is a complex operation as it encompasses many rules, protocols and standards. A framework for secure live migration of VM is proposed. In general, the migration is done in two phases as Live (Continuous execution at the source) and Non-live migration (the execution is suspended). Here, the migration is of VM occurs among the service providers and a policy based VM migration is designed with the cost of communication (Request, Reply, Path establishment) and migration (actual migration and the down time). Finally, the communication cost associated with the VM is analyzed in detail. It is concluded that the downtime is reduced in the parallel migration compared to that in the serial migration strategy. The energy consumption is an important factor in VM allocation these days. Migration from one data center to another itself consumes a lot of energy. Then, migration among the service providers must also be considered, which is not considered in this case [30].

A strategy-proof mechanism for the Heterogeneous PMs Resource Management (HPMRM) has been formulated. That is, multiple VM instances are allocated form heterogeneous PMs. Most of the existing virtual allocation mechanisms are done through single-mapping, which does not provide efficient utilization of multiple types of resources in the cloud. Thus, a multi-mapping mechanism is followed that allocates all the VMs of one user to the PM for VM provisioning and Allocation. After the problem formulation, Vickrey-Clarke-Groves (VCG) - based optimal mechanism is designed for solving the HPMRM problem. Then, the approximation mechanism for mapping the relation between the user-requested VMs and the VM of PMs

is designed. It is concluded that this method provides the space for the users to reveal their actual requests and thereby building a good environment [31].

A complementary VM (CompVM) allocation scheme in cloud computing is proposed. The complementary VM is the VM's whose resource demand is equal to the host capacity. In other words, VM that require high CPU utilization and less memory and the VM that require low CPU utilization form a full utilization of resources. To perform this, first the resource utilization pattern of VMs are identified and the requirement of that particular resource is obtained. The VMs that are complementary to each other are allocated to the same PM. One can assume different VMs as complementary VMs, but to find it accurately two different mechanisms are put forth as: utilization variation based mechanism (short-term jobs and long-term jobs) and correlation coefficient-based mechanism. Also, the resource utilized by the VM reaches the capacity of the PM. This method works only when the VMs have similar task and the resource utilization pattern is predicted. Otherwise, it is not efficient in results [32].

A resource allocation strategy with security-enhanced VM migration is proposed. In any cloud platform, the performance of resource scheduling has a direct impact on the energy consumption and the resource utilization of PM and VM. The inertia and learning factor are dynamically adapted to improve the search performance of Security Enhanced Particle Swarm Optimization (SE-PSO). The fitness function solely depends on the CPU usage. The time-series based predictive exponential smoothing model is used to predict the physical hotspots (i.e., PM) and reduce the VM migration, thereby minimizing the energy consumption. At last, the roulette wheel idea is applied and the long-term optimization of the platform resources is achieved. The PSO implemented in this approach is general and also performs slowly. Therefore, though this approach provides the best results in terms of energy consumption, it is time consuming [33].

A summary of the objectives of all the approaches is given in Table III. We have studied many VM allocation algorithms in this survey. The three important objectives focused by almost all the papers are load balancing, resource utilization and energy efficiency.

Table III: Evaluation Parameters/Metrics of All VM Algorithms

Method	Workload Balancing	SLA Violation	Fault Tolerance	Resource Utilization	Power/Energy Aware
2-ACO [29]	P	P	NP	P	P
SE-PSO [33]	NP	NP	P	P	P
NN [26]	P	NP	NP	P	P
Ensemble [27]	P	NP	P	P	NP
Firefly IPSO[25]	P	P	NP	P	NP
GA-SVM [23]	P	NP	NP	P	P
PEFT-ACO [24]	P	P	P	NP	NP
SC-PSSF [22]	NP	P	NP	NP	P
PPRGear [28]	P	P	NP	NP	P
HPMRM [31]	NP	P	P	P	P
CompVM [32]	P	P	NP	P	P
P- Present; NP-Not Present					

D. Comparison Analysis of VM Allocation Algorithms

We have presented some of the recent methods implemented to resolve the issue of load balancing and VM allocation in the cloud computing environment. Voluminous heuristic, Meta heuristic, machine learning and other approaches are discussed in the above sections. Each of these algorithms has its own advantages and disadvantages which are detailed in Table IV. Many machine learning approaches are used to perform the VM allocation and have shown better results. The drawbacks of machine learning algorithm are listed as follows:

- Though provide high level of accuracy, they are still time consuming.
- High susceptibility error and also require accurate training.
- Requires plenty of resources to bring out a near optimal result in VM allocation.

In addition to these machine learning algorithms, several Meta heuristic approaches are also used for VM allocation. Survey shows that they provide better and optimal results. The disadvantages of Meta heuristic algorithm are given as follows:

- Not much suited for local search and has huge search space while selecting the appropriate PM to map the VM.
- Though it provides optimal solution, it consumes lot of time for deciding the suitable PM to migrate the VM.
- Static conditions are not suited for diverse types of VM demand by the PM since each VM differs in configuration.

Hence, to overcome all these issues hybrid algorithms are developed. The main advantage of is that the objective function

are constructed by considering various parameters. Besides these approaches, there are several other approaches that work using some assumptions and algorithms to provide an efficient VM allocation strategy to a particular extent.

TABLE IV: VM ALLOCATION ALGORITHMS

Algorithm	Description	Pros	Cons
Meta Heuristic Algorithms			
2-ACO [29]	ACO optimally select the positions of VMs, PMs and switches and then construct path and update best solution.	• Minimized Energy Consumption. • High Resource utilization.	• Random decisions. • Time consuming with number of iterations.
SE-PSO [33]	PSO decides VM queue and find position. Also, during migration find the PM.	• Less SLA violation. • Less use of power .	• No hint of response time. • Smoothing forgets the previous values.
Machine Learning Algorithms			
NN [26]	With so many algorithm for VM allocation. Neural Network is used to select the best method based on environment and evaluation parameters.	• Decide algorithm based on situation. • High job completion ratio.	• Not accurate in classification. • Increase complexity with rise in algorithms.
Ensemble Learning [27]	Separate delay sensitive and insensitive VM and map them to appropriate hosts.	• Delay sensitive tasks are allocated first. • Only suitable host.	• Unknown response time. • Computation complexity is high.
Hybrid Algorithms			
GA and SVM [23]	The fitness function of GA which minimizes the overloading problem is cross-validated using the SVM classifier.	• Achieve load balancing. • Simple to manage as only two states.	• Does not work well when VM increases. • Training the dataset is time consuming.
PEFT and ACO [24]	The early finish rate of the VM is predicted and given to the ACO to reach the optimal solution fast.	• Search space is confined. • Less delay and cost.	• Not optimal in all cases. • Ant move only in one direction.
Firefly and IPSO [25]	The firefly algorithm will form the initial allocation of VM and the IPSO is responsible for the reallocation.	• Efficient VM allocation. • High response time.	• Increased bandwidth. • High energy consumption.
Other Approaches			
SC-PSSF [22]	The server which is selected previously is given the priority and the VM is allocated to it.	• High security. • Less use of energy.	• Does not support for different tasks. • Poor load balancing with increased VM.
PPRGear [28]	Based on the overutilized or underutilized sate of the host, the VM is migrated in or out.	• Efficient VM migration. • Performance is also considered.	• Uncertain results with preferred gear. • Increase in the host with underutilized state.
HPMRM [31]	Optimal multi mapping scheme for the VM to PM and also considered the several different types of VM (Heterogeneous).	• Handle any type of VM requests. • Consistency in resource utilization.	• No efficient result for the VM arriving at random. • Time consuming process.
CompVM [32]	Two different VM can be combined to the same PM and achieve the maximum utilization of resources.	• Increased energy efficiency. • High resource utilization.	• Did not focus on cost . • Low response time.

III. Discussion and Future Direction

Cloud computing is consumer oriented and provides services as per their demands. We have analyzed the two most important research areas as task scheduling and VM allocation process in cloud computing in this comprehensive survey. We have analyzed and present the working approaches that give the nearly optimal results. In most of the articles, the mentioned approach is not extended for real-time application due to certain drawbacks. Many different algorithms and procedures are put forth in this survey. We also tabulated the various parameters and other metrics in detail.

While performing the process of task scheduling, the focus must be more on the parameters considered. These metrics must be chosen according to the environmental condition in which it is being implemented. For instance, if it is a large-scale environment energy will be having the least priority and vice versa. This has an impact during the trade-off between parameters. When designing the objective function with respect to the state of the environment, all significant parameters must be included, i.e., deadline, task size and type, execution time and so on.

In VM allocation, the main concern of most of the cloud service providers lies in user satisfaction. Even though VM migration will consume energy, it is necessary to prevent the physical host from being loaded with the work. The number of VM migration is reduced in some papers but still there is certain amount of wastage in energy. Also, most of the approach seen here are not scalable. They work for a particular set of VMs in the environmental setup mentioned in the respective article. When the number of VMs increases, the approach will not work well at the initial allocation stage itself. In some case, VM running on a host overtakes the capacity of the PM and results in huge workload with enormous wastage of power.

Besides, the future directions in the task scheduling and VM allocation of cloud computing are: (1) identify and execute the delay sensitive tasks with minimum execution time at first, (2) implement multiple queue that orders the task based on the type of their application to give equal priority to all kinds of task. For instance, order all the tasks that demand real-time services in one queue and those tasks that demand non-real-time services in a separate queue or tasks can be queued based on their deadline constraint, etc., (3) design a scheduler that meets the SLA requirements, (4) reduce number of the VM migration by accurate allocation mechanism. (5) Deep learning approaches are becoming current trend in the VM allocation process. Therefore, while framing hybrid algorithm, this method is used for classification of the demands from the PM while it is requesting for VM (6) design of the VM allocation strategy that is suitable for real-time application.

IV. Conclusion

In this comprehensive survey, we have described the current trends in the various implementation of task scheduling and VM allocation of cloud computing. The major areas of concern are task scheduling and VM allocation. Large volume of tasks arrives from the users in random in an instant. These tasks are scheduled and the right task is given the priority. Then, PM is given requested the VMs by the cloud based on their source requirement of the task in such a way that it does not lead to overprovision or under provision of VM resources. Both these operations are implemented using several approaches and techniques by considering various assumptions and in different environments. Also, these both operations have several significant parameters which are crucial to obtain an optimal result. In this survey article, we have analyzed the algorithms that are implemented to solve these both issues and found that algorithms with single objective function (considering limited parameters) will not yield the best results. Energy consumption and maximum CPU utilization are the important objectives in addition to the user satisfaction. Also, it must ensure the minimum rate of SLA violation. From the survey, it is clear that the many standard scheduling techniques are improved with innovative approaches including the heuristic and Meta heuristic algorithms. It is evident that if these approaches are combined in the right way by considering multiple parameters as well as reducing the time and algorithmic complexity, then powerful procedure can be developed. In most cases, instead of combining two Meta heuristic algorithms, other types of hybridization will give best results at a reasonable time period. Eventually, task scheduling and VM allocation can be done in an efficient manner.

References

[1] B. A. Al-Maytami, P. Fan, A. Hussain, T. Baker, and P. Liatsis, "A task scheduling algorithm with improved makespan based on prediction of tasks computation time algorithm for cloud computing," *IEEE Access*, vol. 7, pp. 160916-160926, November 2019.

[2] P. Srivatsava, and R. Khan, "A review paper on cloud computing," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 8, no. 6, pp. 17-20, July 2018.

[3] A. Rashid, and A. Chaturvedi, "Cloud computing characteristics and services: A brief review," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 2, pp. 421-426, February 2019.

[4] Md. Haris, and R. Z. Khan, "A systematic review on cloud computing," *International Journal of Computer Sciences and Engineering*, vol. 6, no. 11, November 2018.

[5] A. Belgacem, K. Beghdad-Bey, and H. Nacer, "Task scheduling in cloud computing environment: A comprehensive analysis," *International Conference on Computer Science and its Applications, Advances in Computing Systems and Applications,* Springer, vol. 50, pp. 14-26, August 2018.

[6] A. R. Arunarani, D. Manjula, and V. Sugumaran, "Task scheduling techniques in cloud computing: A literature

survey," *Future Generation Computer Systems,* Elsevier vol. 91, pp. 407-415, February 2019.

[7] R. K. Jena, "Energy efficient task scheduling in cloud environment," *Energy Procedia*, Elsevier, vol. 141, pp. 222-227, December 2017.

[8] R. N. Parasar, "Optimization of virtual machines in cloud environment," *International Journal of Advance Research, Ideas and Innovations in Technology*, vol. 5, no. 3, pp. 1768-1770, 2019.

[9] S. K. Panda, and P. K. Jana, "Load balanced task scheduling for cloud computing: A probabilistic approach," *Knowledge and Information Systems*, Springer, vol. 61, no. 3, pp. 1607-1631, December 2019.

[10] B. H. Malik, M. Amir, B. Mazhar, S. Ali, R. Jalil, and J. Khalid, "Comparison of task scheduling algorithms in cloud environment," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 9, no. 5, pp. 384-390, 2018.

[11] A. P. Sheetal, and K. Ravindranath, "Priority based resource allocation and scheduling using artificial bee colony (ABC) optimization for cloud computing systems," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 6S, April 2019.

[12] P. Krishnadoss, and P. Jacob, "OLOA: Based task scheduling in heterogeneous clouds," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 1, pp. 114-122, February 2019.

[13] Sreelakshmi, and S. Sindhu, "Multi-objective PSO based task scheduling - A load balancing approach in cloud," *1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, IEEE, April 2019.

[14] F. Yiqiu, X. Xia, and G. Junwei, "Cloud computing task scheduling algorithm based on improved genetic algorithm," *3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, IEEE, March 2019.

[15] A. Gupta, and R. Garg, "Load balancing based task scheduling with ACO in cloud computing," *International Conference on Computer and Applications (ICCA),* IEEE, September 2017.

[16] C. Zhang, P. Luo, Y. Zhao, and J. Ren, "An efficient round robin task scheduling algorithm based on a dynamic quantum time," *International Journal of Circuits, Systems and Signal Processing*, vol. 13, 2019.

[17] A. Al-Rahayfeh, S. Atiewi, A. Abuhussein, and M. Almiani, "Novel approach to task scheduling and load balancing using the dominant sequence clustering and mean shift clustering algorithms," *Future Internet*, MDPI, vol. 11, no. 5, May 2019.

[18] A. Keivani, F. Ghayoor, and J. R. Tapamo, "A review of recent methods of task scheduling in cloud computing," *19th IEEE Mediterranean Electrotechnical Conference (MELECON)*, IEEE, May 2018.

[19] S. B. Alla, H. Benalla, A. Touhafi, and A. Ezzati, "An efficient energy-aware tasks scheduling with deadline-constrained in cloud computing," *Computers*, MDPI, vol. 8, no. 2, pp. 1-15, June 2019.

[20] A. Adhi, B. Santosa, and N. Siswanto, "A meta-heuristic method for solving scheduling problem: Crow search algorithm," *International Conference on Industrial and System Engineering (IConISE)*, vol. 337, 2017.

[21] V. M. Sivagami, and K. S. Easwarakumar, "An improved dynamic fault tolerant management algorithm during VM migration in cloud data center," *Future Generation Computer System*, vol. 98, pp. 35-43, November 2018.

[22] H. Jiaa, X. Liua, X. Dia, H. Qia, L. Conga, J. Lia, and H. Yanga, "Security strategy for virtual machine allocation in cloud computing," *International Conference on Identification, Information and Knowledge in the Internet of Things, IIKI,* Elsevier, vol. 147, pp. 140-144, January 2019.

[23] G. Singh, and M. Mahajan, "A green computing supportive allocation scheme utilizing genetic algorithm and support vector machine," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 9S, July 2019.

[24] A. Kaur, and B. Kaur, "Load balancing optimization based on hybrid Heuristic-Metaheuristic techniques in cloud environment," *Journal of King Saud University – Computer and Information Services*, Elsevier, March 2019.

[25] M. M. Golchi, S. Saraeian, and M. Heydari, "A hybrid of firefly and improved particle swarm optimization algorithms for load balancing in cloud environments: Performance evaluation," *Computer Networks*, Elsevier, vol. 162, October 2019.

[26] J. N. Witantoa, H. Limb, and Md. Atiquzzamanc, "Adaptive selection of dynamic VM consolidation algorithm using neural network for cloud resource management," *Future Generation Computer Systems*, Elsevier, vol. 87, pp. 25-42, October 2018.

[27] R. Shaw, E. Howley, and E. Barrett, "An intelligent ensemble learning approach for energy efficient and interference aware dynamic virtual machine consolidation," *Simulation Modelling Practice and Theory*, Elsevier, October 2019.

[28] X. Ruan, H. Chen, Y. Tian, and S. Yin, "Virtual machine allocation and migration based on performance-to-power ratio in energy-efficient clouds," *Future Generation Computer Systems*, Elsevier, vol. 100, pp. 380-394, November 2019.

[29] A. S. Chakravarthy, Ch. Sudhakarand, and T. Ramesh, "Energy efficient VM scheduling and routing in multi-tenant cloud data center," *Sustainable Computing:*

Informatics and Systems, Elsevier, vol. 22, pp. 139-151, June 2019.

[30] S. K. Addya, A. K. Turuk, A. Satpathy, B. Sahoo, and M. Sarkar, "A strategy for live migration of virtual machines in a cloud federation," *IEEE Systems Journal*, IEEE, vol. 13, no. 3, pp. 2877-2887, September 2019.

[31] X. Liu, W. Li, and X. Zhang, "Strategy-proof mechanism for provisioning and allocation virtual machines in heterogeneous clouds," *IEEE Transactions on Parallel and Distributed Systems*, IEEE, vol. 29, no. 7, pp. 1650-1663, July 2018.

[32] H. Shen, and L. Chen, "CompVM: A complementary VM allocation mechanism for cloud systems," *IEEE/ACM Transactions on Networking*, IEEE, vol. 26, no. 3, pp. 1348-1361, June 2018.

[33] T. Ma, C. Xu, Z. Zhou, X. Kuang, and L. Zhong, "SE-PSO: Resource scheduling strategy for multimedia cloud platform based on security enhanced virtual migration," *15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, IEEE, June 2019.