Resource Allocation in Cloud Computing

Chapter · August 2019

CITATIONS
7

READS
7,989

3 authors:

Vivek Kumar Prasad
Nirma University
49 PUBLICATIONS **263** CITATIONS
SEE PROFILE

Anuja Nair
Nirma University
25 PUBLICATIONS **137** CITATIONS
SEE PROFILE

Sudeep Tanwar
Nirma University
558 PUBLICATIONS **14,647** CITATIONS
SEE PROFILE

Chapter 10: Resource Allocation in Cloud Computing

Vivek Kumar Prasad[1], Anuja Nair[2], and Sudeep Tanwar*[3]

[1,2,3] Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad (Gujarat), India, 382481.
E-mails: vivek.prasad@nirmauni.ac.in[1], anuja.nair@nirmauni.ac.in[2], sudeep.tanwar@nirmauni.ac.in[3]

Abstract- Cloud computing is one of the important utilities in the present era of the technological world, where the multitenant usage the resources available at cloud with the help of the services such as Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). The resources should be made available to the end user with minimal management and an efficient resource allocation mechanism has to be adopted in a way, to avoid the situations of over-provisioning and under-provisioning. As cloud also serves the mechanism of elasticity. So, static scheduling algorithms cannot be a part of the cloud workload scheduling techniques and as result, dynamic scheduling will play an important role for efficient utilization of the resources available in the cloud. The dynamism of the resource availability and the tasks allocated can be calculated by making use of various Particle swarm optimization (PSO) algorithms. The resources should be made available with respect to service level agreement (SLA) decided in between the end users and cloud service providers (CSP) at the time of establishing the services request. Various policies and metrics with respect to the SLA will be used to offer resources to the end-users. The benefits of the proper utilization of the resources can increase the revenue of the CSP.

Keywords: *Service level agreement, cloud service providers, Resource Allocation, Policy, Scheduling.*

Learning Outcomes

After reading this chapter, the students will be able to:

- To identify the importance and need for resource allocation in cloud computing.
- To identify the various policies used for resource allocation.
- To identify the various scheduling algorithms designed for the resource allocation.
- To identify the performance criteria of scheduling algorithms used for resource allocation.

10.1 Introduction

The augmented consciousness of energy consumption in data centres' has enthused the practice of dynamic management of VM's in the servers. Such cloud environments where applications have dynamic and variable requirements, the capacity management as well as the demand prediction are an essential tool to manage the resources.

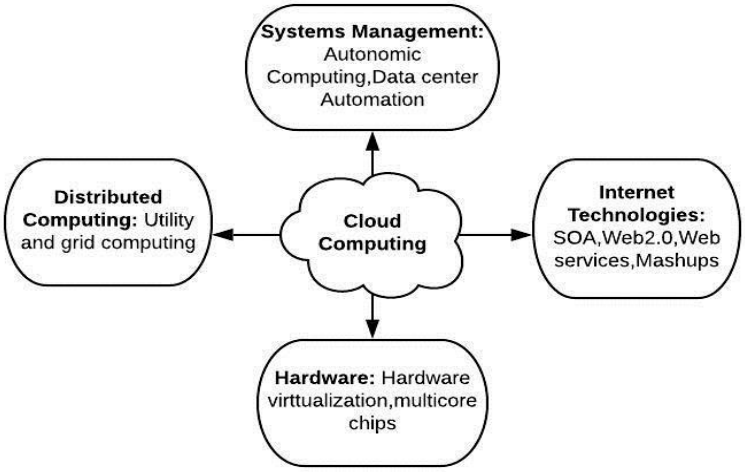


Figure 10.1: The chief advent of Cloud Computing

Fig. 10.1 shows the major chief advent of cloud computing. In this chapter, we will focus on the two subject i.e Hardware and System Management with respect to resource allocation in cloud computing. These particulars expose the possibility of delivering computing services with reliability and speed with the benefits of cloud resource management in terms of utilization and scalability of the cloud resources. The businesses will earn revenue by utilizing these local machines.

Better management of the agreed SLAs and energy ingestion in data centers requires the practice of dynamic resource consolidation in Virtual Machines(VMs) to physical machines at a periodic interval of time [1]. The machines that are in an idle state can be turned off and can be put into the low power consumption state and at the same time, the overloaded and overheating condition can be evaded. In literature, there are multiple Virtual Infrastructure (VI) managers are available which includes the services of dynamic resource distribution structures that uninterruptedly monitor the consumption of the resources in the data centers and also reallocates the resources as per the demand of the end users and the available resources.

10.2 Importance of Resource Allocation

Cloud computing is the utility similar to water, gas, and electricity, where the users use these and pay as per their usages.CC is an emerging research infrastructure which inherits its base from virtualization technology, grid computing and service-oriented architecture (SOA). The CC offers Infrastructure as a Service (IaaS) based on pay as you go manner and on-demand resources figuring models as per the requirements of the user's demand.

Lets us now understand the importance of the resource allocation in the era of CC. It’s a process of allocating an accessible resource to the applications hosted by the clients into the cloud. If the resources are not allocated/ managed properly in a good manner then the services starve. The resource provisioning manager solves the aforementioned problem by permitting the service provider to accomplish the resources for each of the discrete components by using various resource allocation strategy discussed in the next section.

10.3 Strategies for Resource Allocation

The strategies of resource allocation [2] can be defined as the mechanism for obtaining the guaranteed VM and/or physical resources allocation to the cloud users with minimal resource struggle, avoiding over, under-provisioning conditions and other parameters as shown in Fig. 10.2, This needs the amount and its types of resources required by the applications in order to satisfy the user's tasks, the time of allocations of the resources and its sequels also matters in case of the resource allocation mechanism.

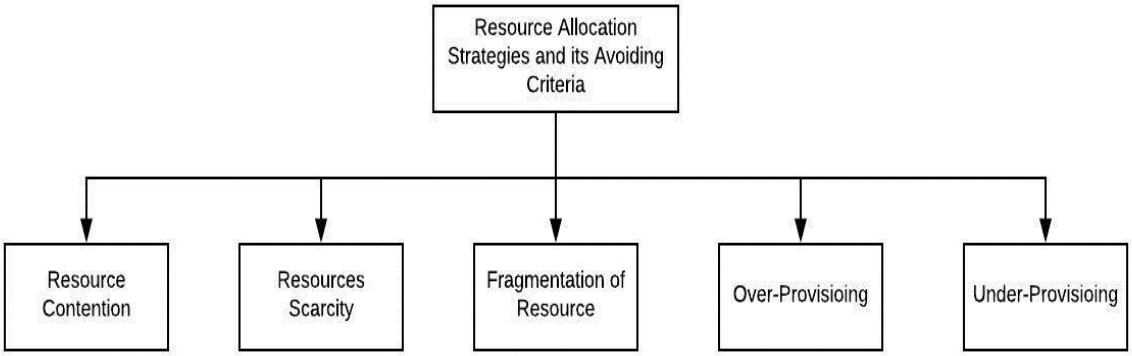


Figure 10. 2: Resource allocation strategies and its avoiding criteria

Let's discuss these criteria of avoidance one by one

- Resource contention condition ascends when two applications claim to access the same resource at the identical time.
- Shortage of resources arises when there are inadequate resources.
- Resource fragmentation condition arises when the resources are out-of-the-way. [There will be adequate resources but not enough intelligent to allocate to the desired application.]
- Over-provisioning of resources happens when the said tasks get spare resources than the actual demand.
- Under-provisioning of resources befalls when the user's tasks are allotted with fewer quantities of resources than the actual demand.

Advantages and disadvantages of resource allocation strategies:

Advantages:-

- The major advantage of resource allocation is that the client neither has to install H/W nor S/W to get the requests and to host the request over the cloud.
- The client does not require to spend on hardware as well as software systems.
- The Cloud service providers (CSP) can share the resources on the internet during the resource shortage.
- There is no bound of medium and global place. We can spread our requests and submissions anywhere across the globe.

Disadvantages:-

- As users lease these resources of cloud from remote servers to for their work and they do not have control over these resources
- The profound knowledge is required for managing and allocating the resources in the CC, as entire information about the working of the cloud environment depends upon the cloud service provider (CSP).

- In the deployment model such as public cloud, the end users data can be vulnerable to phishing attacks and hacking etc. As the servers on the cloud are open and interconnected, this will be so easy for malware to spread in the network.
- Migration issue arises, when the client wants to move to another cloud provider for the better services or storage of their data. It's a difficult and time-consuming process to transfer vast data from one provider to another.

10.4 Resource Allocation Policies and Algorithms

In this section, we will realize the various policies [3] and algorithms [4] associated with the Resource Allocation Strategy (RAS).

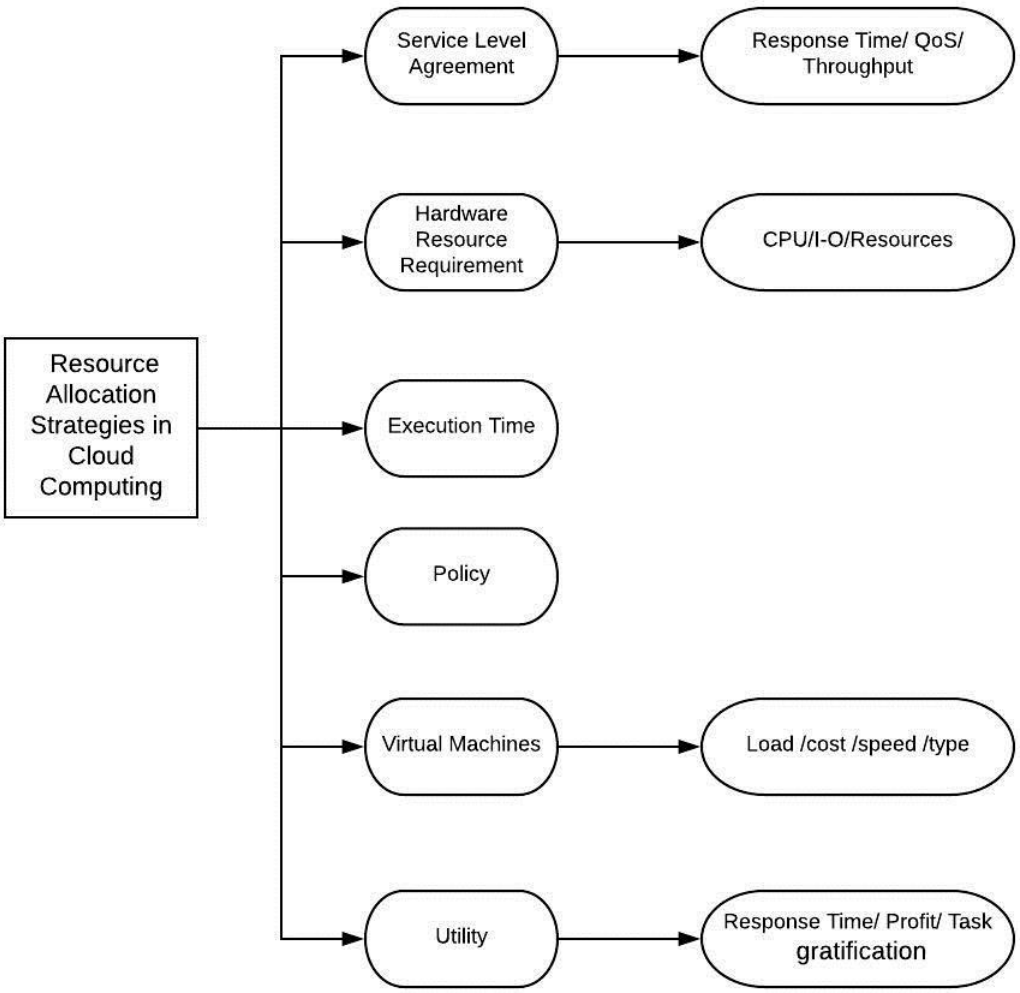


Figure 10.3: Resource allocation in cloud computing and its strategies

Fig.10.3 shows some basic parameters associated with the resource allocations in cloud computing and its strategies which works with different types of services, application type with various resource demand and infrastructure. The following points will discuss the same.

Service Level Agreement (SLA): SLA is the agreement done in between the end user and cloud service provider before the communication starts. SLA with respect to the service of cloud, such as a Software as a Service (SaaS), the main consideration here is the quality of service parameters of the cloud service provider's side, like a load on the current environment and cost associated with the data centre.

Hardware Resource Requirement: This is associated with the term called hardware resource utilization as per the application's requirement. To achieve the same scheduler is used. The tasks can be classified w.r.t memory bound, Disk I/O bound, CPU-bound etc. Table 10.1 shows various applications and their resources requirement in the cloud environment.

Applications types in cloud environment	Example	Resources Required/related to the associated QoS parameters
Websites	Social networking and in-formational sites.	Huge Storage and high bandwidth
Application of science	Bio-informatics, numerical analysis, data analysis	High computational capability
e-commerce	Shopping and banking transaction	Flexible Computing load, such as during holidays times there may be more requests from the users.
Finance services	Insurance and banking	High availability and security
Mobile services	Mobile applications	High availability
Software required for business	SCM, CRM, and ERP	Availability

Table 10.1: The application and its required resources in CC

According to the categories aforementioned, the resources will be allocated. As an example, Nimbus, Eucalyptus, and open nebula are open source framework used for resource management in the cloud.

Execution Time: is the time required for executing the tsk in cloud.Estimating the execution time of the tasks in cloud computing plays an important role while dealing with a proactive approach for monitoring and predictions. This can also be used for workflow scheduling in the cloud era.

Policy: To get the optimality in resource utilization and load balancing results, certain conditions and parameters have to be considered and for the same, the policies can be used. As an example, the policy can be when the load increased (more than 80%) the load will be transferred to another

server. Usually, the virtual machine policy can be used for the enhancement of resource utilization.

Virtual Machines: The cloud system is composed of the various virtualized network, operating system, and resources with the capability of live migration across various cloud infrastructure. Dynamic allocation of the resources use these concepts to scale up/ scale down according to the demand of the end users. Similarly, the virtualized computation environment robotically moves across various infrastructures (different cloud deployment models) and balance its resources to maintain the cost, speed etc and follows the policy of non-preemptible scheduling algorithms.

Utility: The objective function can be used for managing the resources (VMs) in a dynamic fashion such as meeting its QoS terminology, a function for less cost with respect to performance etc. These objective functions are known as utility, which can be analyzed based on the profit, target set, response time and performance, which is explained in the next section.

10.4.1 Performance-based RAS

The performance can be achieved by[5] :

- The dynamic distribution of the resources such as CPUs, memory etc to meet its QoS objectives
- Allocating the request to the high priority tasks
- Profit-based resources distribution algorithms
- Assigning job with respect to less cost.

Use policy grounded heuristic procedures in the occurrence of conflicting goals by adjusting the parameter discussed in Table 10.2, which indicates few parameters that need to be tuned for achieving the performance in RAS.

These data can also be used in intelligence scheduling techniques, which results in the wise decision steps to allocate the resources to the associated tasks. In the same way, these parameters can be tuned for monitoring and predict the resources in reactive and proactive ways. The next sections describe the same.

Sl. No	Parameters
1	Selecting the best threshold point under-provisioning
2	Selecting the best threshold point for over- provisioning
3	No. of resource consumers
4	No. of registered cloud consumer
5	Cost for under-provisioning
6	Cost for over- provisioning
7	Total Usages of resources
8	Consumed unit
9	Unconsumed unit
10	Cost of resources per unit

Table 10.2: Parameters for performance in RAS

10.4.1.1 Locality-Aware Task Scheduling

To accomplish the highest utilization and throughput of resource utilization in a cloud environment, the resource scheduler system must make awfully fast scheduling judgments. As

huge tasks are becoming data intensive and undergoing data explosion. These tasks encompass the processing of enormous data at various working nodes in the distributed systems. Hence data locality, data-aware, and load balancing are indispensable requirements [6].

Data locality: In a distributed system the data-intensive computing, resources need for every node may not be equally divided (uniform). The critical cause for this uneven distribution of the jobs is because of the term called a data locality and this also affects the execution time of the tasks. Preceding studies show the data locality distresses the throughput of Hadoop jobs knowingly, and they recover the locality by transferring the tasks to the other nodes that have enough computation to handle the corresponding data. In other words, to advance locality, when the job is planned, the figuring node with the corresponding data do have accessible computing niches to let the job done. If a computing niche is unavailable, the job must be scheduled to a distant (remote) node, which requires the transfer of essential data to the alternate node.

10.4.1.2 Reliability Aware Scheduling

With the rising gauge of cloud computing usages, the network failures and other performance related issues have become inevitable. To achieve reliability we need to focus on designing reliability-aware task scheduling algorithms. This makes use of directed acyclic graph (DAG) for tasks dependency.

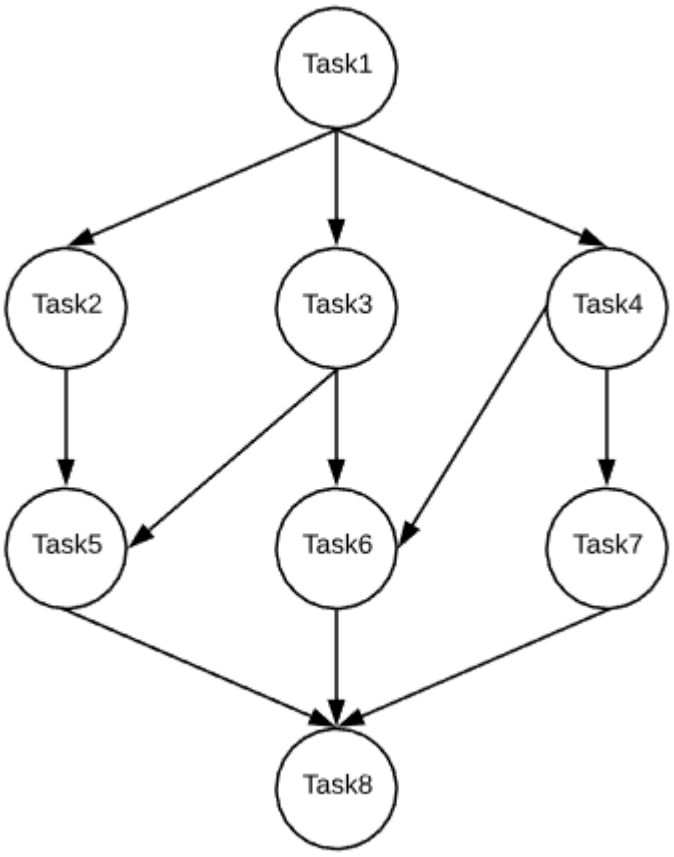


Figure 10.4: Task graph for parallel application

DAG: As shown in Fig. 10.4, the directed acyclic graph (DAG) is a depiction of a parallel claim. In a DAG, the nodes signify application tasks and the directed edges signify the inter-task addictions, such as precedence limits. The task scheduler work is to assign the tasks of a claim to

the respective processors in order to achieve the satisfaction related to the precedence requirement and least makespan can be attained. This is also an NP-hard problem. Hence various heuristics can be used to achieve the goals of the scheduling techniques. The task scheduler can be classified into various parts based upon there depicted properties as shown in Fig. 10.5.

The reliability analysis block will figure out the processor's reliability with respect to different frequency in order to maintain the reliability of the system. Finally, the scheduler will schedule the user's task based upon the DAG and system's reliability criteria.

Please note that it has been assumed that the application is a parallel one and every user has provided the information about the tasks. The main objective of the scheduler is to map the user's task to the processors with a minimum schedule length.

The common properties of the scheduler have been described below:-

Task Priorities Stage. This stage is vital for listing out the scheduling algorithms in cloud

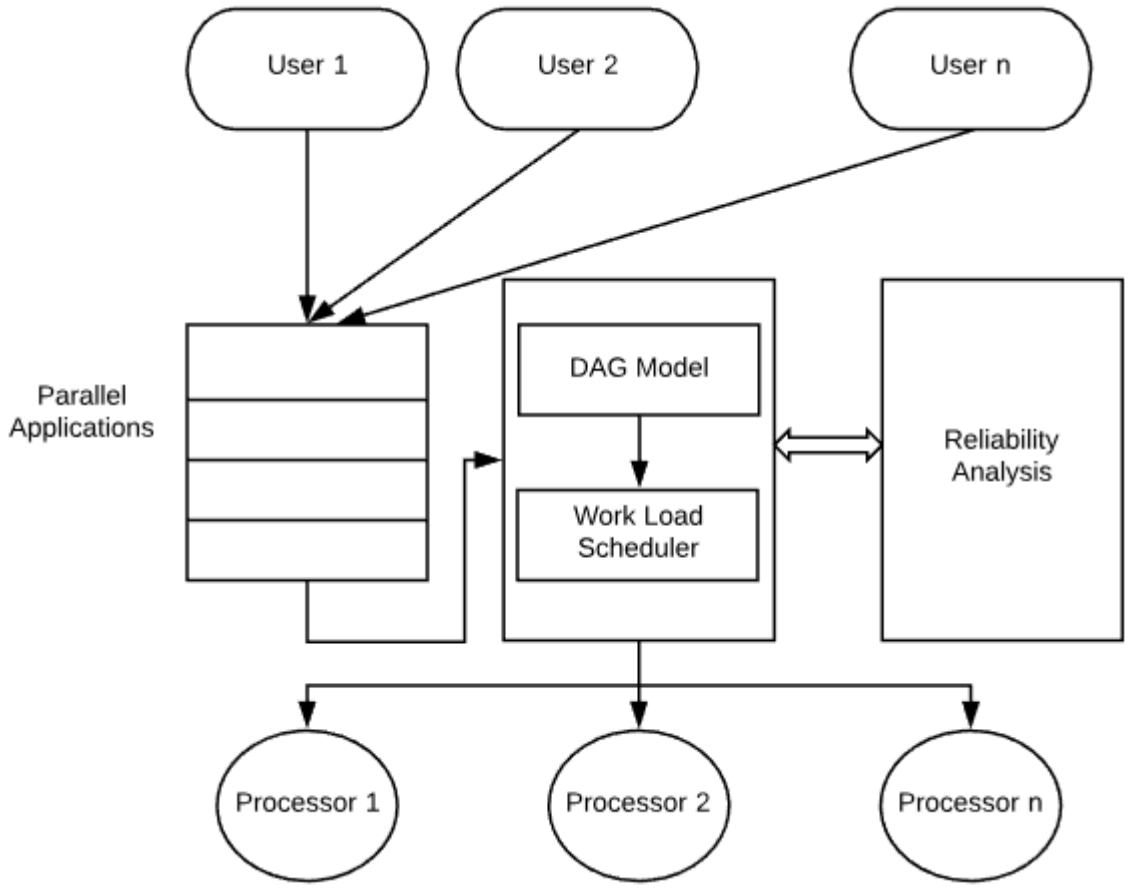


Figure 10.5: Reliability aware task scheduling architecture

environment. The task dispensation list will be created by arranging the tasks by decreasing order of some pre-defined rank utility.

Task Assignment stage:- In this stage, the jobs will be assigned to the processors with the following parameters:

- Earliest execution finish time
- High reliability

Note:- The parameters can be added with respect to the various criteria of reliability, for example, we the criteria if fault tolerance, then the parameter can be the average downtime, mean time to repair the fault etc.

10.4.2 Cost Based RAS

In cloud computing consists of computing resources such as RAM, CPU, memory, bandwidth etc. which will be shared by multiple users. The virtualization techniques will be used by which multiple users can share these shared resources. The cloud computing is a cost-based business model where the users pay as they use the said resources [7]. The goal of the cloud service provider is to generate the profits with its user's gratification. The billing will be generated based on the size and time of the resources used by the end users.

This section now describes the cost based resource allocation strategy which will lessen the total completion time, execution time and better profit in terms of revenue with higher user satisfaction level. To achieve the desired aforementioned concept, this technique will migrate the Virtual Machines from one physical host to an alternative physical machine to avoid the conditions of under and overload circumstances of the hosts. The following important criteria have to be full-filled in order to achieve the cost based RAS

- Do proper load balancing to deliver the services appropriately.
- The Quality of the services should not be degraded with respect to some of the metrics and its associated policy and its threshold value.
- Optimize the provider's profits and user's satisfaction.

To achieve the above criteria the important component is keeping an eye on the total energy consumption of the cloud which is subject to the total number of lively servers. There are so many works done on this approach where we minimize the number of the less utilized running server to another (large) server till this fully utilizes which is called as server consolidation. As this large server will reduce the time required for the execution. Also reduces the energy. On the other hand, the aim of load balancing is to enhance the completion time, response time, memory provisioning, network bandwidth etc.

10.4.2.1 Energy Aware RAS

Data centers munch hefty amount of electricity. As per the data circulated by HP, 100 server racks can devour 1.3 MW of power and another 1.3 MW are needed by the cooling systems, thus this adds the cost of USD 2.6 million for each year. These cooling systems also produce CO_2 which is not suitable for the green sustainable computing environment.

Apart from this the applications which have been optimized and the management of resources in a dynamic manner can consequently improve utilization and also diminishes energy consumptions in data centers of the cloud. As a counter-example of this, we can achieve this by

consolidating the workloads (tasks) onto the smaller number of servers and shutting off the idle resources.

While designing the architecture of the cloud following points can be considered:-

- The algorithms must follow the SLAs while doing energy-aware resource allocation and
- Provisioning.
- To achieve energy efficiency the energy-aware and autonomic algorithms have to be considered such as self-management in case of dynamic changes which are happening while allocating and deallocating the resources.
- To identify the procedures for energy-efficient cloud computing which intelligently maps VMs to the physical machines

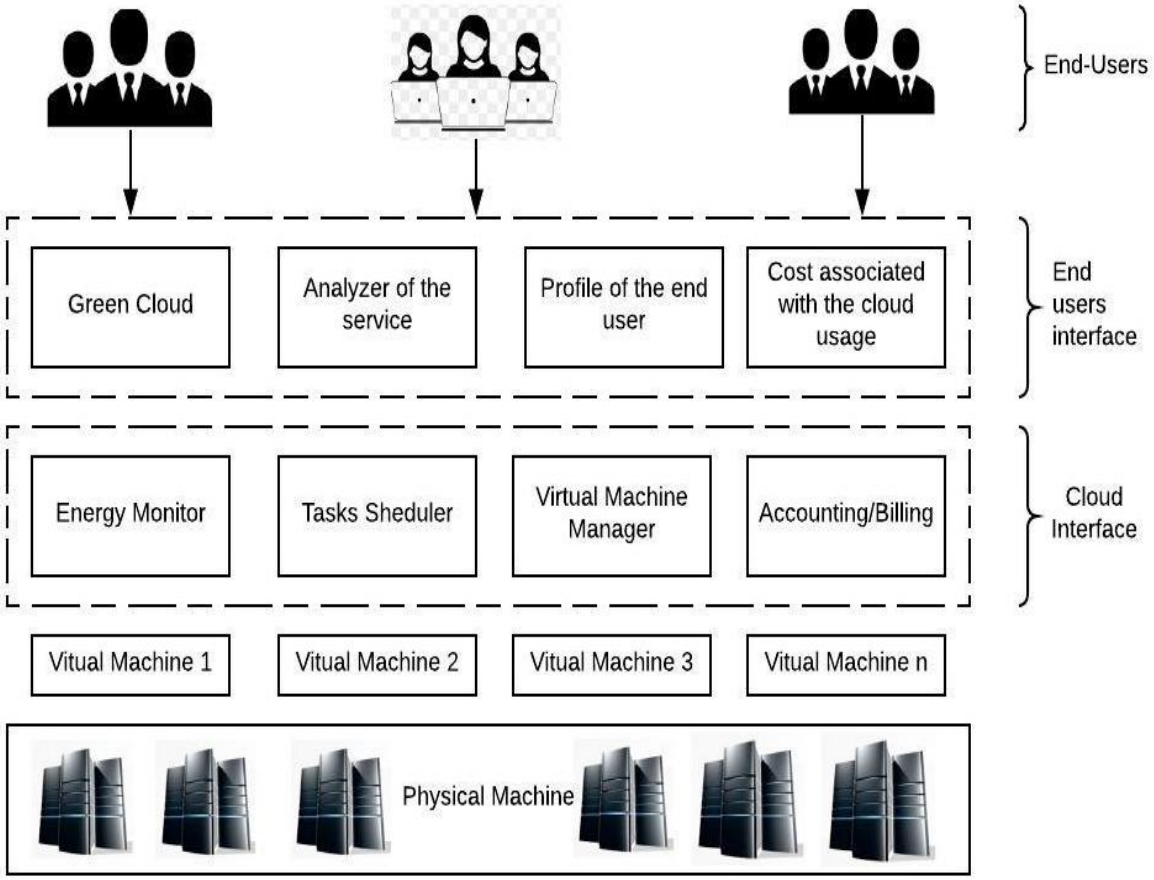


Figure 10.6: Architecture for the energy-aware resource allocation

Let's discuss each of the components used in Fig. 10.6.

End-User:- they can submit the assignment from anywhere of the globe, the thing that requires focus here is that the application which is requested by the end-users have to be analysed, such as

if the end user wants to deploy web-based application with varying workload as per the number of users associated with that web application.

Green Cloud:- It's an interface between the end user and infrastructure of the cloud. This negotiates with the end user as per the SLAs, metrics, policies, and QoS and acts in terms of schemes related to energy saving.

Analyser of the services: Analyzes the resources requirement with respect to the current demand by keeping eyes on current load and information about the energy.

Profile of the end user: need to assign the special privileges and priority to the specific end users Cost associated with the cloud usages: Chooses how requests of the services are billed to achieve the demand and supply of figuring resources and enable the prioritizing facility distributions successfully.

Energy Monitor: Detects energy ingesting triggered by physical machines and VMs then delivers the evidence to the VM manager to act in an energy-efficient manner / to make good resource allocation decisions.

Task Scheduler: Allocates requests to VMs and regulates resource prerogatives for the billed VMs. If the auto scaling functionality has been entreated by the end user the this also decides when the VMs are to be removed or added to meet the claim.

VM Manager: Retains track of every accessibility of VMs and their resource utilization. This also invokes the charge of transferring VMs across physical machines as well as provisioning of new VMs to familiarize the settlement.

Accounting/Billing: Identifies the actual usage of the resources and its associated cost. The historical information of the resource consumption can be reused to improve the allocation decisions

VMs: Manifold VMs can be dynamically stopped and started on a solitary physical machine according to received requirements, henceforth providing the elasticity of arranging numerous partitions of resources on the equivalent physical machine to dissimilar wants of service, the request is an important term. Manifold VMs can concurrently lane requests grounded on different operating system surroundings on a solitary physical machine.

Physical Machines: The physical machines provide the hardware infrastructure for generating virtualized resources to meet facility requirement and demands.

10.4.2.2 Software as a Service Layer RAS

To convey the hosted services to the clients, the SaaS provider has to maintain their own set of infrastructures to maintain and rent this from the cloud providers of the infrastructures. As a result, the SaaS provider has to tolerate an extra cost and to minimize the cost incurred with respect to

the cloud resources; the SaaS provider has to minimize the resources cost by maintaining the minimum service level to the clients. This section discusses the various approach used for resource allocation done by the SaaS provider with minimized infrastructure and SLA defilements.

Hence, to attain the SaaS provider's intentions of maximizing profit and customer consummation levels, following questions can be answered.

- How to face the heterogeneity of the infrastructural level of the cloud?
- How to draw the relationship between infrastructural parameters and customer requirements?
- How to achieve the dynamic changes incorporated because of the customer requests?

Hence, while thinking about the resource allocation w.r.t SaaS; following points has to be considered.

- Design the algorithms by keeping views of the customer's SLA's and QoS parameters.
- Need to map carefully the demand made by the client and the available infrastructure.
- Design and instrument the scheduling algorithms in such a way so that this leads to profit (by reducing the infrastructural cost) and minimizing SLA violations.

The scheduling appliance controls which (and where ?)the type of VM has to be originated by integrating the heterogeneity of VMs in terms of their cost, dynamic facility beginning time, and time of data transfer. As well as; this also manages to lessen the experienced consequences for managing the dynamic service's demand, when the customers are partaking (sharing) the resources. The SLA properties parameters have been mentioned in Table 10.3 along with the description.

Sl.No	**Properties of the SLAs**	**Description**
1	Request type	This identifies the customer request category, as first-time rent payment (asking for new services)or this may be upgraded facilities (upgraded product) .
2	Product type	The product offered by the SaaS as standard, enterprise and professional product (this contains orders, sales, accounting functions and report functions)
3	Account type	This addresses about the thoroughgoing quantity of accounts the client can have such as group, department, and team.
4	Contract length	For how much time the software system is lawfully available for the client for usages.

5	Number of accounts	The genuine figure of accounts that a client requires to generate. (Should be ≤ the extreme amount of accounts for specific account type)
6	Number of records	The maximum records, a customer can acquire during the transaction which influences the data transfer time throughout the service updates. (These values will be available predefined in SLA)
7	Response time	This indicates the elapsed time amongst the beginning of the service and end of a demand for software service.The violation arises when the genuine elapsed time extends than the predefined one in SLA

Table 10.3: Properties of SLAs

The properties of resource allocation to assure SLAs is working properly has been discussed in Table 10.4.

Sl.No	**Properties of the SLAs**	**Description**
1	VM types	What are the various kinds of VM available (the three kinds of VM can be: Large, Medium and Small)
2	Service initiation time	The Total time is taken for initialization of the VM, which is deployed with a software product.
3	VM price	The total cost generated by SaaS provider for providing VM to the client's demand per unit of time. Which included the following:- Power Equipment (Physical) The price of administration Network
4	Data transfer time	Time is taken to transfer the gigabyte information/ data from one virtual machine to another virtual machine.
5	Data transfer speed	The data transfer speed is subject to; the distance, location and the performance of the network.

Table 10.4: Properties of SLAs with respect to resource allocation strategy

The delay can be calculated by observing the differences between the actual SLA parameters and the present observations. There are other algorithms too; that make usages of the QoS parameters such as penalty rate, service initiation time and the arrival rate from the perspective of both the SaaS provider as well as the customers.

10.4.3 Performance and Cost based RAS

CC is an striking computing prototypical; as this allows for the provision and de- provisioning of resources on-demand (elasticity). Such a procedure of allocation and reallocation of resources with the view point of performance and cost are the key to house unpredictable demands and improve the return on investment (RoI) from the infrastructure supported by the Cloud; Again how to achieve these point has been discussed below.

10.4.3.1 Workflow Scheduling

The IaaS layer cloud consists of various VMs in the pool of available physical resources. Scheduling these VMs as per the requirement of the uses demand is an important and complex requirement, which requires careful analysis and intelligence. There are various well know workflow scheduling algorithms are available in the literature [8] and few important algorithms were discussed here. Basically, most of the algorithms have been analyzed on the approached mentioned in Table 10.5.

Sl. No	Approaches Analyses
1	Time of Execution
2	Cost of Execution
3	Time of communication
4	Deadline constraint
5	Cost of communication
6	Budget Constraint
7	Hours invested on the Wasted partial instance.
8	Delay in VM startup.
9	VM type
10	Failure of VM or Tasks.
11	Hybrid cloud resources.

Table 10.5: Workload scheduling approaches

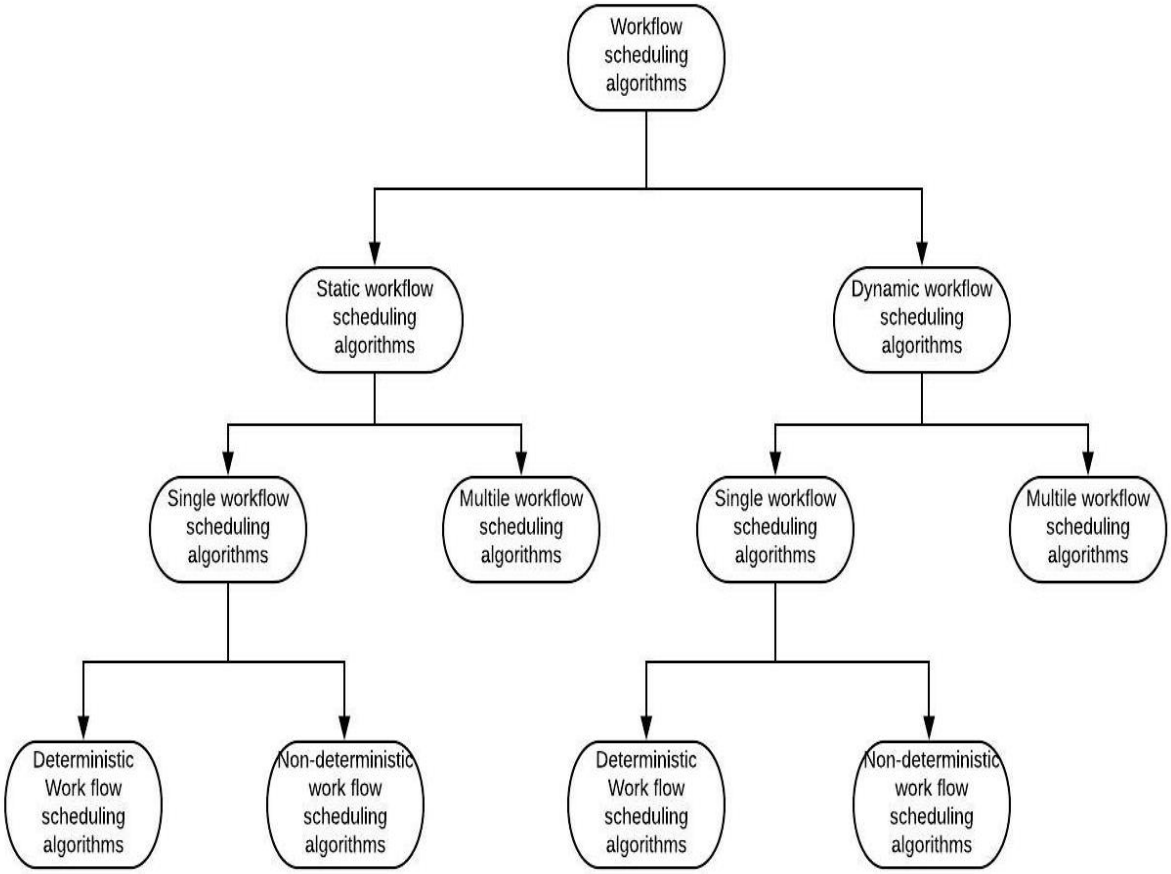


Figure 10.6: The workflow scheduling algorithms classification

Figure 10.6 shows the classification of the workflow scheduling algorithms techniques, where this has been classified as static and dynamic workflow algorithms, then these techniques can be again refined as single workflow scheduling algorithms and multiple workflow algorithms.

Single workflow scheduling algorithms: Scheduling the single workflow, with the deterministic and non – deterministic structure [9] with respect to DAG(Directed Acyclic Graph).

Multiple workflow scheduling algorithms: Scheduling the multiple workflows which can be dependent or independent of each other.

As the cloud is elastic in nature and dynamic workflow is an obvious technique to be followed, which facilitates the altering resources extents at runtime. The conditional requirement of the workflow observed, the corresponding resources can be released or added at runtime on the ultimatum. Hence the cloud computing is used as an operative accelerator to deal with the runtime resource allocation on claim/ demand and maintaining the QoS constraint too for changing workflow scenarios. This will finally progress the application's completion time and deal with the unexpected circumstances. In limitation, the situation where the fixed resources can lead to a poor concert.

Scheduling techniques: Advances resources reservation in cloud computing.

In case of advanced resource reservation, the resources will be reserved in advanced or pre-reserved. So that there will be the availability of a guaranteed resource for certain period of time.

As in case of managing a private cloud which devours restricted resources and an instantaneous provisioning prototypical is insufficient.

Note: the reservation and queuing are needless when the resources are already available to satisfy the needs of the end user.

Research direction for future:

- To identify the scheduling techniques in terms of coherent data exchange, load balancing task migration for workflow applications.
- How to provision cloud-based resources scheduling techniques in terms of deadline constraint and within their budget.
- To balance the users of the cloud and resources which are locally available in such a way that scheduling objectives can be met.
- Need to match intelligently the workflow requirements and any of the available workflow scheduling criteria as well as the cloud service provider's capabilities.

10.4.3.2 Algorithm Scheduling Ant Colony Optimization (ACO)

The scheduling algorithms can be identified w.r.t to their important characteristics as mentioned in Table 10.6.

Sl.No	Characteristics	Description
1	Targeted System	The scheduling algorithms developed for the specific system, this can be a cloud computing, heterogeneous system or may be a grid.
2	Criteria of Optimization	The metrics specified are cost and makespan which are used by the schedulers for making the decisions.
3	Awareness about the multicore	While scheduling the system can be checked for multicore, that can be considered for selection of the resources.
4	On request resources	The resources can be acquired on the leased basis for long-term or on requirement basis (on demand). These on-demand resources can be acquired by making use of scheduling algorithms for the workflow.
5	Resources that are reserved	Resources reservation algorithms should be considered to reserve the resources for a long time.
6	SLA	SLA organization should be structured hierarchically for scheduling the algorithms, this organized structure will allow the clients as well as the provider to interact and negotiate prices and capacities of the resources.
7	Heuristic information	Heuristics data are some problem-based standards; which is used to identify the search track and based

		on the identified problem of the workflow, we need to design the heuristics here. The main heuristics are mentioned below • Makespan: Used for approximating the max. time of completion, by using the concluding time of the up-to-date task; as all tasks are scheduled. In case the makespan of the cloudlet is not curtailed then the demand will not be accomplished on time. • Cost: The cost here indicates the payment identified against the usages or utilization of the resources. This is paid to the cloud service provider by the end users of the cloud. Here the determination is to increase the revenue/profit for CSP and reducing the expenses of cloud users with effective operations • Multicore: Multicore processing resources. • Bandwidth / Throughput: Throughput practices the deliberation of entire number of tasks, which are successfully implemented. In the era of cloud computing, throughput means some tasks accomplished in a convinced time period.

Table 10.6: Characteristics of the scheduling algorithms

The motivation behind the Ant colony optimization is:-

- To come up with a scheduling algorithm that should be made in such a way that this maximizes the performance of the computational cloud w.r.t clients inclinations.

ACO is an optimization (metaheuristic combinatorial optimization technique) that works on the foraging behavior of the Ants. As shown in Figure 10.7, the ACO jolts with the initialization of the parameters and the permissible range the ant can travel and these results into the construction path. These paths are then classified as a decision variable and then the function of the objective has to be analyzed. Next step is to evaluate whether the optimum solution has been reached or not. If the answer is yes then stop, otherwise the best path is chosen from the available evaluated values and the pheromone updates will be carried out from the best path, to form the latest novel set of allowable series; for the subsequent iterations.

For Instigating ACO for any algorithms related to scheduling, the following steps have to be identified

- Pheromone Initialization
- Heuristic information initialization

- Ant’s random generation
- Path mapping of ants
- Objective function’s evaluation
- Updating the pheromone

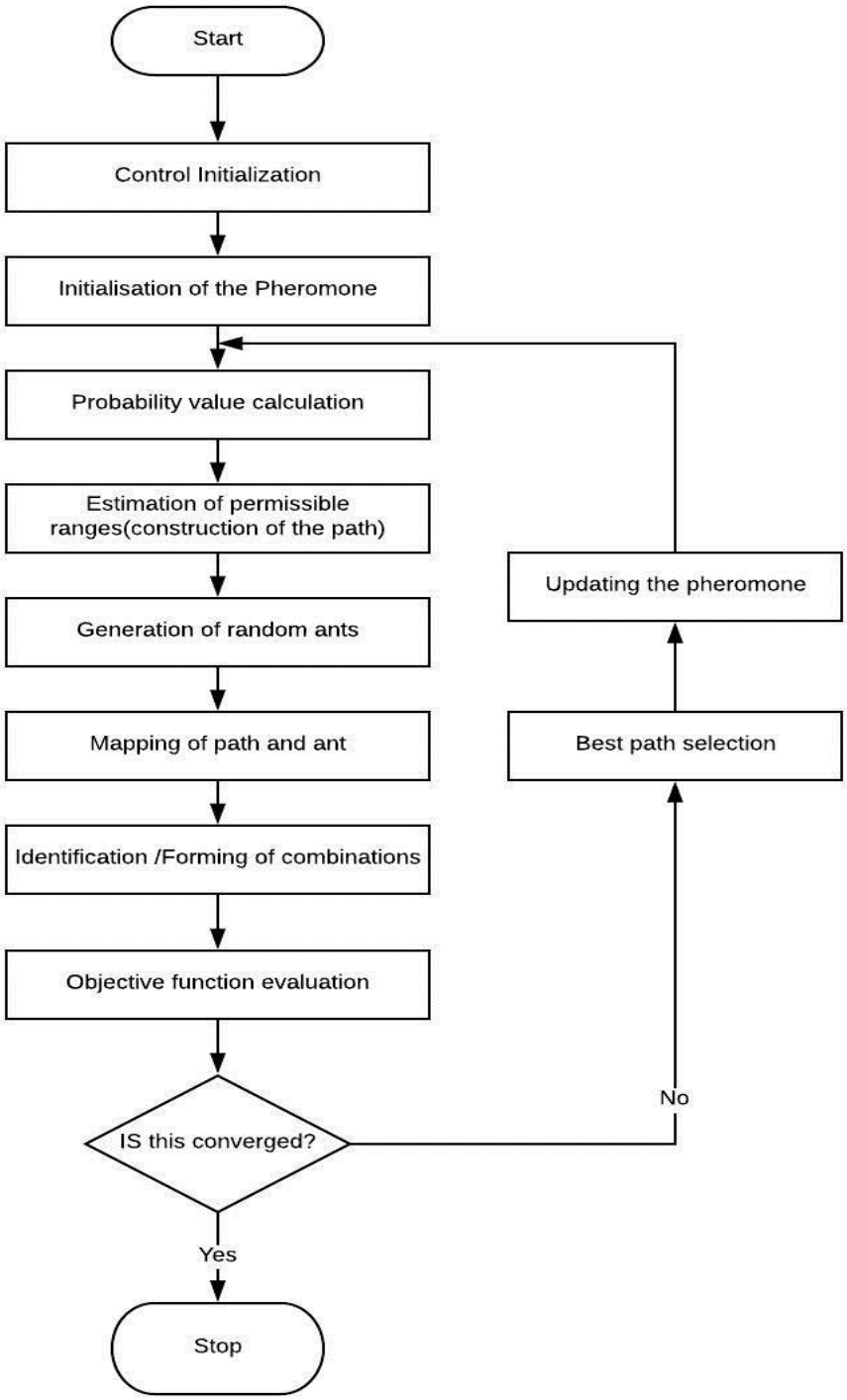


Figure 10.7: ACO working Model

ACO optimizes the scheduling workflow in the cloud using various heuristics data like makespan, cost, multicore, and bandwidth. Hence to prove that this practice to be operative; this can be compared with other optimization techniques of Particle Swarm Optimization (PSO), which has been discussed in the next section.

10.4.3.3 Algorithm Scheduling Particle Swarm Optimization (PSO)

The cloud users will pay the charged amount based upon the pay per use basis. The user's job may experience large data retrieval or the cost of execution as and when these are scheduled based on execution time. In addition to this to optimize the execution time and cost involved for doing a data transfer between the available resources has to be considered. Hence the particle swarm optimization based scheduling algorithms can be used.

The PSO is a global exploration based self-adaptive optimization practice. This is similar to the other population grounded procedures such as genetic algorithms; this works upon the social behavior of the particles. Every particle regulates its trajectory with respect to the (local) best position and the (global) best particle of the complete population. This results in the stochastic idea of the particle and congregates to minima (global) with a judicious better solution.

There are a wide range of applications with stumpy computational cost for PSO, which has been mentioned below:-

- The responsive voltage regulator schemes.
- Mining of data
- Chemical Engineering
- Area of pattern recognition
- Environmental Engineering

The PSO has also been used to resolve the problems of

- NP-Hard problematic issues related to task allocation and scheduling.

Hence PSO can be used to improve:-

- The model for task resource mapping and also to minimize the whole cost of accomplishment.
- To design a heuristic that customs the usages of PSO to unravel task resource mappings grounded (based) on the proposed model requirement.

As PSO is a bio-inspired procedure which is purely based on population. The procedure begins after the significance of the available population of the candidate solutions. The particle is the term, which has been derived from the population of the individual elements; for these particles, the fitness value is deliberated. These details (PSO Algorithm) has been mentioned in Figure 10.8.

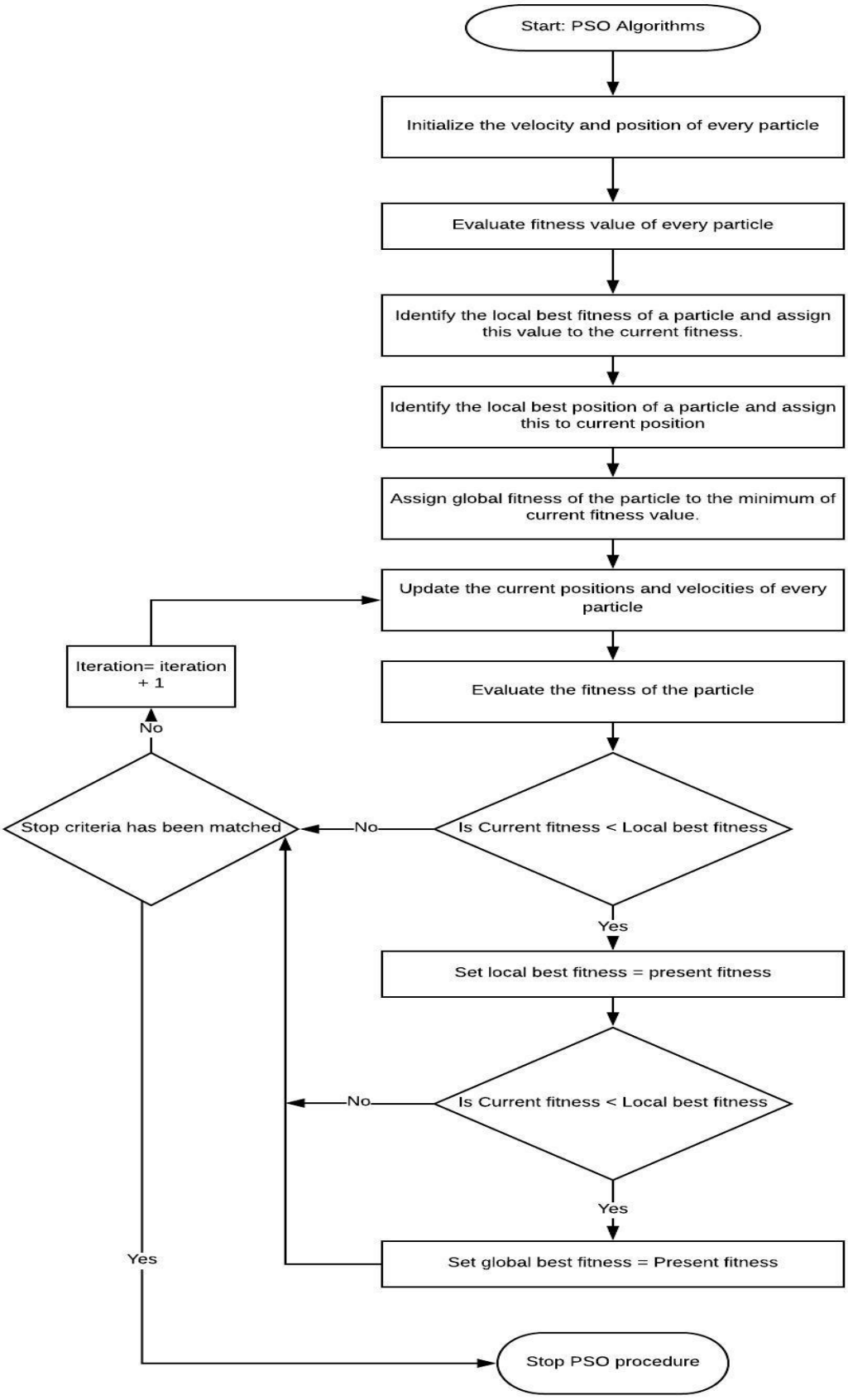


Figure 10.8: PSO Algorithm flow chart

Steps for PSO algorithms

Step #1: Need to set the dimension of the particle equal to the size of the ready task.

Step #2: Randomly initialize the particle position and velocity too.

Step #3: For every particle identify its fitness value.

Step #4: If the fitness value is healthier than the prior best value; set the present fitness value as the new

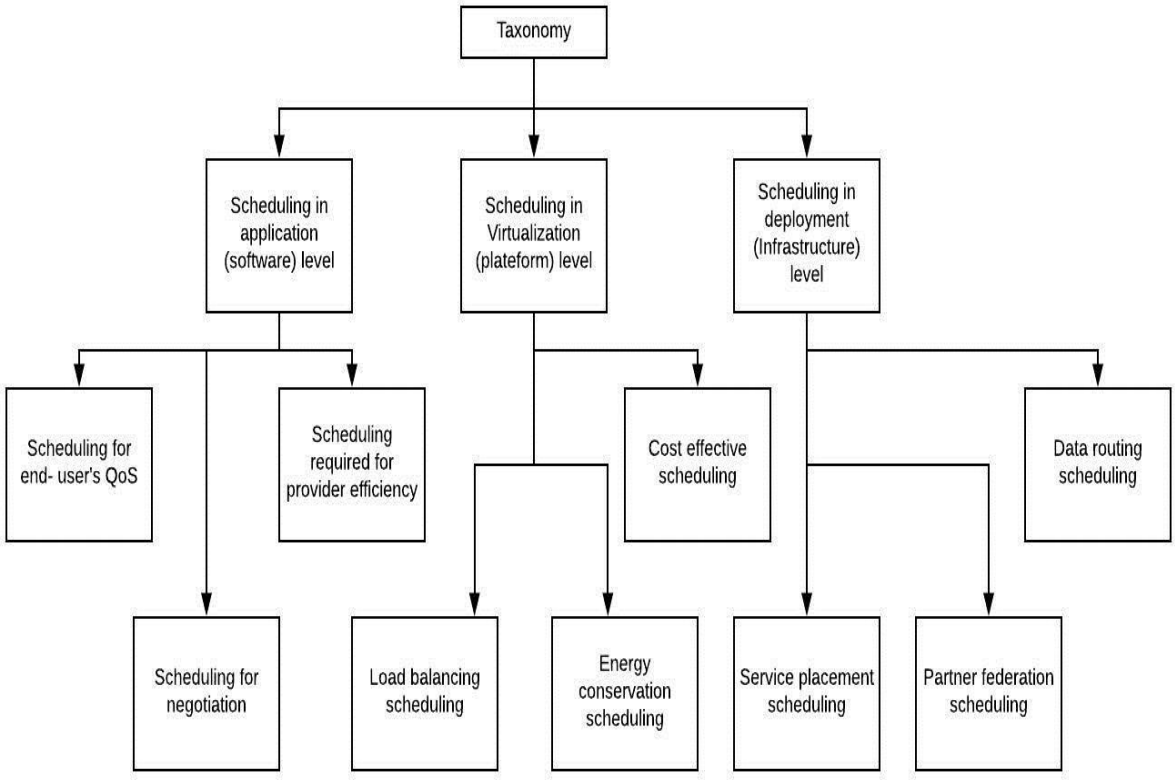
Step #5: After completion of the 3rd and 4th steps for every particle, now select the best particle.

Step #6: For every particle, now calculate the velocity and update its position

Step #7: If the stopping principles or maximum iteration is not satisfied; then repeat from step 3 onwards.

10.5 Cloud resources levels and scheduling nomenclature

When managing resources will come into the picture of a cloud computing environment; the scheduling can be classified with respect to the services layers as discussed in Figure 10.9.



. *Figure 10.9: Cloud resource scheduling taxonomy*

Scheduling in application level: scheduling the user applications to the physical or virtual resources with optimal efficiency and QoS.

Scheduling in virtualization level: Give emphases on mapping resources (virtualized) to the physical resources with finest energy conservation and load balancing techniques.

Scheduling in deployment level: Give emphases on optimal, outsourcing, placement of the services, data centers, data routing, migration of applications and strategic infrastructure.

Whether we use a Genetic algorithm (GA)[10], ACO or PSO; These evolutionary algorithm techniques will follow a cycle of

(i) Evaluation of fitness

(ii) Selection of candidate

(iii) Variation of trial

These above-mentioned algorithms are non-deterministic and a posteriori in nature, which indicates that they do not require prior guidance, can be suitable for complex scenarios and multiobjective solutions in the scheduling of cloud resources.

The comparison of the scheduling objectives with respect to use's QoS and its related work has been shown in Table 10.7. It shows how the research has begun from GA to, model formulation, objective extension and algorithms enhancement. Also reflects the association of growth from GA to ACO and then to PSO. The general comments on strength, objectives, experimental environments (test bed),scales and results of the algorithms are listed and also compared here.

Algorithm	**Objectives**	**Strength/Limitations**	**Environment and experiments**	**Test Bed scale**	**Result compared**
GA [11]	Makespan	Logic consists of numerical simulation and no cloud environment has been considered.	Numerical based simulation	2 tasks 2 resources	NA
GA [12]	Makespan, cost, distance,bandwidth ,reliability	Uses local search and combines weight of QoS parameters. No cloud environment has been used	Numerical based simulation	20 tasks 8 resources	NA
GA[13]	Cost and makespan	Both cost and makespan has been combined and allowed to use the GA adaptive strategy.	CloudSim	50 resources 50 tasks	Only cost/makespan
GA[14]	Availiability, reputation	This considered the network resources,database and VM; also	Numerical based simulation	3 to 20 resources 6-120 tasks	Random selection and exhaustive search

	,cost and makespan	usages different QoS (weights) to achieve the objectives			
GA[15]	Cost and makespan	Models the scheduling algorithms in case of multiobjective problem of optimization.	Amazon cloud	2 to 128 resources , 25- 1000 tasks	GA and PSO
ACO[16]	Makespan	Adds the time slot data to the pheromone to optimize the makespan, this doesn't consider the fault tolerance issues.	MS live mesh, Google app engine	25 different kinds of tasks	Google resource allocators/Traditional ACO microsoft
ACO[17]	Security, cost,makespan and reliability	Uses various objectives by weights according to the user's QoS, but doesn't explains any comparisions	Numerical based simulation	10-50 resources ,10 tasks	NA
ACO [18]	Makespan	Priority of the task has been considered ; also allows users to define there objectives	CloudSim	8 resources , 20-100 tasks	Random distribution algorithms
PSO [19]	Cost	This consideres both the data transmission and cost of computation; here the PSO encode does not deliberate the resources differences.	Amazon cloud	3 resources 5 tasks	**BRS** algorithm
PSO [20]	Cost	Make use of randomized adaptive greedy search to initialize the solution which are feasible	Amazon cloud	3- 20 resources , 50-300 task	Traditional PSO-BRS algorithms
PSO [21]	Cost ,reliability and makespan	Usages set based PSO (discrete) and only one objective	Numerical based simulation	6-10 resources , 9-120 tasks	Markov decision process and ACO

		is considered every time			
PSO[22]	Cost	Uses makespan as deadline constraint ; considered communication cost and computational cost	CloudSim	6 resources , 50- 1000 tasks	SCS, IC-PCP
PSO [23]	Cost	Make use of renumber(strategy) to make particles learn eficiently	Numerical based simulation	10 resources , 200 tasks	PSO
PSO [24]	Cost	Make use of renumber(strategy) ; extends to multiobjective model of scheduling	Numerical based simulation	10 resources , 100 tasks	Renumber, PSO

Table 10.7: Comparisons of various scheduling approaches for cloud users QoS

Scheduling Criteria: With a view point of provider's effectiveness.

Here, in this section we will discuss the scheduling algorithms of the cloud resources with the view point of provider efficiency. In this context the basic scheduling objectives can be included as:

(i) Load balancing
(ii) Utilization maximization
(iii) Energy consumption minimization

The related work has been summarized as per their objectives (optimized) in Table 10.8.

Objectives	**Algorithms**	**Strengths/ limitations**	**Test beds**	**Scale of the experiments**	**Compared results**
Load balancing	GA[25]	Usage memory balance and CPU	Numerical based simulation	20 resources ,100 tasks	Mim-min algorithm
	ACO[26]	This is an distributed algorithms ; where ant detects load(heavy) and transfers some load to light load node	NA	4 resources	NA

Utilization maximization	ACO[27]	Ant chooses weighty load resources; no experiments and comparisons	NA	30 resources	NA
	ACO [28]	PSO avoids prematurity; ACO selects the resources	Numerical based simulation	50-400 tasks	Traditional ACO
Energy Consumption	GA[29]	To guide the evaluation this uses shadow price strategy	Numerical based simulation	10-50 resources, 500 -5000 tasks	Traditional GA
	GA[30]	GA improved with native search	Hadoop	200 resources,450 tasks	Hadoop MapReduce
	ACO[31]	Cuckoo search used as heuristics and ACO used as a framework	Xen	1-10 resources,1-256 tasks	Traditional ACO
Multi objectives	GA [32]	Profit, carbon emission and consumption of energy.	Feitelson's PWA	NA	Greedy scheduling

Table 10.8. Comparison on scheduling methods for provider point of effectiveness

Conclusion:

The cloud computing requires resource scheduling techniques for the management of the resources and also focuses on the profit generated by the cloud service provider. The scheduling can be classified with respect to the three services of the cloud (SaaS, PaaS, and IaaS) and can be identified into three categories; application layer scheduling, virtualization scheduling and deployment layer scheduling. For each layer of scheduling criteria, certain points has to be considered such as current workload, scheduling objectives with different viewpoints such as; cloud user concerned objectives, system concerned objectives and cloud provider concerned objectives. Then evolutionary computing algorithms have been discussed, such as Gentic algorithms, Ant colony optimization and particle swarm optimization.The various challenges and research direction of the future, shceduling in real time, large scale scheduling and dynamic adaptive scheduling has been discussed in this chapter.The reaserch in this area is in the infancy stage; still new problems will emerge with advancement of Big data and internet of things in cloud comuting.

References

[1] Patel, P., Ranabahu, A. H., & Sheth, A. P. (2009). Service level agreement in cloud computing.

[2] An, B., Lesser, V., Irwin, D., & Zink, M. (2010, May). Automated negotiation with decommitment for dynamic resource allocation in cloud computing. *In Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*: Volume 1 (pp. 981-988). International Foundation for Autonomous Agents and Multiagent Systems.
[3] Nathani, A., Chaudhary, S., & Somani, G. (2012). Policy based resource allocation in IaaS cloud. *Future Generation Computer Systems*, 28(1), 94-103.
[4] Aggarwal, R. (2018). Resource Provisioning and Resource Allocation in Cloud Computing Environment.
[5] Lee, H. M., Jeong, Y. S., & Jang, H. J. (2014). Performance analysis based resource allocation for green cloud computing. *The Journal of Supercomputing*, 69(3), 1013-1026.
[6] Balagoni, Y., & Rao, R. R. (2017). Locality-Load-Prediction Aware Multi-Objective Task Scheduling in the Heterogeneous Cloud Environment. *Indian Journal of Science and Technology,* 10(9).
[7] Singh, A., Juneja, D., & Malhotra, M. (2017). A novel agent based autonomous and service composition framework for cost optimization of resource provisioning in cloud computing. *Journal of King Saud University-Computer and Information Sciences*, 29(1), 19-28.
[8] Rimal, B. P., & Maier, M. (2017). Workflow scheduling in multi-tenant cloud computing environments. *IEEE Transactions on Parallel and Distributed Systems*, 28(1), 290-304.
[9] Gupta, I., Choudhary, A., & Jana, P. K. (2017, November). Generation and Proliferation of Random Directed Acyclic Graphs for Workflow Scheduling Problem. In Proceedings of the 7th *International Conference on Computer and Communication Technology* (pp. 123-127). ACM.
[10] Schram, M., Kerbyson, D. J., & de la Torre, L. (2018, March). Towards Efficient Resource Allocation for Distributed Workflows Under Demand Uncertainties. In Job Scheduling Strategies for Parallel Processing: *21st International Workshop*, JSSPP 2017, Orlando, FL, USA, June 2, 2017, Revised Selected Papers (Vol. 10773, p. 103). Springer.
[11] H. Zhao, S. S. Zhang, Q. F. Liu, J. Xie, and J. C. Hu. 2009. Independent tasks scheduling based on genetic algorithm in cloud computing. *In Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing*, 1–4.
[12] G.N. Gan, T. L. Huang, and S. Gao. 2010. Genetic simulated annealing algorithm for task scheduling based on cloud computing environment. *In Proceedings of the International Conference on Intelligent Computing and Integrated Systems*. 60–63.
[13] J.W. Ge and Y. S. Yuan. 2013. Research of cloud computing task scheduling algorithm based on improved genetic algorithm. *In Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering*. 2134–2137.
[14] Z. Ye, X. F. Zhou, and A. Bouguettaya. 2011. Genetic algorithm based QoS-Aware service compositions in cloud computing. Database Systems for Advanced Applications, *Lecture Notes in Computer Science*, Volume 6588. Springer, Berlin, 321–334.
[15] C. Szabo and T. Kroeger. 2012. Evolving multi-objective strategies for task allocation of scientific workflows on public clouds. *In Proceedings of the IEEE World Congress on Computation Intelligence.* 1–8.
[16] S. Banerjee, I. Mukherjee, and P. K. Mahanti. 2009. Cloud computing initiative using modified ant colony framework. *World Academy of Science, Engineering and Technology* 56 (2009), 221–224
[17] H. Liu, D. Xu, and H. K. Miao. 2011. Ant colony optimization based service flow scheduling with various QoS requirements in cloud computing. *In Proceedings of the 1st ACIS International Symposium on Software and Network Engineering*. 53–58.
[18] L. N. Zhu, Q. S. Li, and L. N. He. 2012. Study on cloud computing resource scheduling strategy based on the ant colony optimization algorithm. *International Journal of Computer Science Issues* 9, 5 (2012), 54–58.

[19] S. Pandey, L. L. Wu, S. M. Guru, and R. Buyya. 2010. A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments. In Proceedings of the *24th IEEE International Conference on Advanced Information Networks and Applications.* 400–407.
[20] Z. J. Wu, Z. W. Ni, L. C. Gu, and X. Liu. 2010. A revised discrete particle swarm optimization for cloud workflow scheduling. *In Proceedings of the International Conference on Computational Intelligence and Security*. 184–188.
[21] W. N. Chen and J. Zhang. 2012. A set-based discrete PSO for cloud workflow scheduling with user-defined QoS constraints. *In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics.* 773–778.
[22] M. A. Rodriguez and R. Buyya. 2014. Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds. *IEEE Transactions on Cloud Computing* 2, 2 (2014), 222–235.
[23] H. H. Li, Y. W. Fu, Z. H. Zhan, and J. J. Li. 2015a. Renumber strategy enhanced particle swarm optimization for cloud computing resource scheduling. *In Proceedings of the IEEE Congress on Evolution Computation,* in press.
[24] H. H. Li, Z. G. Chen, Z. H. Zhan, K. J. Du, and J. Zhang. 2015b. Renumber coevolutionary multiswarm particle swarm optimization for multi-objective workflow scheduling on cloud computing environment. *In Proceedings of the Genetic Evolutionary Computation Conference.*
[25] K. Zhu, H. G. Song, L. J. Liu, J. Z. Gao, and G. J. Cheng. 2011. Hybrid genetic algorithm for cloud computing applications. *In Proceedings of the IEEE Asia-Pacific Services Computing Conference*. 182–187.
[26] K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh, N. Nitin, and R. Rastogi. 2012. Load balancing of nodes in cloud using ant colony optimization. *In Proceedings of the 14th International Conference on Computer Modelling and Simulation*. 3–8.
[27] Q. C. Lv, X. X. Shi, and L. Z. Zhou. 2012. Based on ant colony algorithm for cloud management platform resources scheduling. *In Proceedings of the World Automation Congress.* 1–4.
[28] X. T. Wen, M. H. Huang, and J. H. Shi. 2012. Study on resources scheduling based on ACO algorithm and PSO algorithm in cloud computing. *In Proceedings of the 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science*. 219–222.
[29] G. Shen and Y. Q. Zhang. 2011. A shadow price guided genetic algorithm for energy aware task scheduling on cloud computers. *In Proceedings of the International Conference on Advances in Swarm Intelligence. Lecture Notes in Computer Science*, Volume 6728. Springer, Berlin, 522–529.
[30] X. L. Wang, Y. P. Wang, and H. Zhu. 2012. Energy-efficient multi-job scheduling model for cloud computing and its genetic algorithm. *Mathematical Problems in Engineering,* Article ID 589243, 1–16.
[31] R. G. Babukarthik, R. Raju, and P. Dhavachelvan. 2012. Energy-aware scheduling using hybrid algorithm for cloud computing. In Proceedings of the *3rd International Conference on Computing Communication & Networking Technologies*. 1–6.
[32] Y.Kessaci,N.Melab, and E.G. Talbi. 2011. A Pareto-based GA for scheduling HPC applications on distributed cloud infrastructures. *In Proceeding of the International Conference on High Performance Computing and Simulation.* 456–462.