Cloud Computing Resource Management

Anwesha Mukherjee, Debashis De, and Rajkumar Buyya

Abstract Cloud computing is a significant and sophisticated paradigm in the field of modern computer technology and services. In cloud computing, efficient resource management becomes crucial since efficient resource sharing is essential to provide good Quality of Service (QoS). This chapter illustrates different approaches to cloud resource management. Along with cloud computing, resource management in edge and fog computing is also discussed. The challenges of cloud resource management are also highlighted.

Keywords Resource management · Energy · Security · Machine learning · Edge/Fog computing

1 Introduction

In the context of Internet-based computing and service provisioning, cloud computing has become a core paradigm that has gained importance in recent years [1]. Cloud computing functions as a pay-per-use model that is built on an on-demand, utility-based consumer-provider service paradigm [2]. It uses both parallel and distributed computing, with virtualized and interconnected computers that are dynamically

A. Mukherjee (✉)
Department of Computer Science, Mahishadal Raj College, Mahishadal 721628, West Bengal, India
e-mail: anweshamukherjee2011@gmail.com; anweshamukherjee@mail.mrc.ac.in

D. De
Centre of Mobile Cloud Computing, Department of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, West Bengal, B.F.-142, Salt Lake, Sector-I, Kolkata 700064, West Bengal, India
e-mail: debashis.de@makautwb.ac.in

R. Buyya
Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia
e-mail: rbuyya@unimelb.edu.au

A. Mukherjee et al. (eds.), *Resource Management in Distributed Systems*, Studies in Big Data 151, https://doi.org/10.1007/978-981-97-2644-8_2

supplied as a single set of computing resources based on service-level commitments between customers and service providers [1].

Cloud computing provides Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [2, 3]. Virtual machines and storage, which include storage and processing, are available through IaaS. PaaS provides a runtime environment and tools for app building and deployment. Users can utilize SaaS to access several applications via a network.

To further emphasize the adaptability of this technology, the deployment types of cloud computing are categorized as public clouds, private clouds, community clouds, and hybrid clouds [2]. Anyone can use public cloud services on pay-per-use basis. Private clouds, on the other hand, only let people inside an organization to use cloud services that are not available to the public. Community clouds offer services to specific communities or groups, while hybrid clouds blend two or more deployment types.

Cloud computing follows a Service-Oriented Architecture (SOA) that is based on distributed computing and virtualization [3]. The cloud is comprised of a multitude of shared resources and a huge number of users, with access to these resources being facilitated by the network. Thus, efficient resource management is a significant issue of cloud computing.

What is Resource Management?

Resource management deals with the allocation and release of resources, and virtualization methods provide resources on-demand and in flexible manner [3]. When a task is received, the task is placed on the virtual machine (VM) that has been allotted to the user. After completion of the task, the procured resources are released. Resource allocation is carried out according to the service level agreement (SLA) established between the supplier of services and the consumer. The SLA contains required service level details of the consumer, information regarding payment process and SLA violation penalty [3]. For resource management in cloud, market-oriented techniques are advised, which can provide resource sharing and on-demand resource provisioning [3]. Resource management mainly deals with resource provisioning, resource allocation, and resource monitoring [4].

What Are the Cloud Resources?

Cloud computing involves the provision of resources to customers as a service, which they can rent via the Internet. The resources are broadly classified as physical resources or hardware resources and logical resources or software resources [2]. As the cloud offers computing as a utility to the consumer on-demand, the cloud resources are classified into five following categories by [2]:

- Fast computation utility
- Storage utility
- Communication utility
- Energy/power utility
- Security utility.

We will discuss about these resources in Sect. 2.

What Are the Objectives of Resource Management?

In cloud computing, usually, all resources are virtualized and shared among multiple consumers. There are several challenges in resource management such as network load, energy efficiency, SLA violation, load balancing, profit maximization, etc. The resource management metrics have multiple challenges, for example, network loads and energy efficiency introduce SLA violations, and reduction in SLA violations enhances energy consumption and affects the profit. Furthermore, resource management methods for public or private cloud environment may not be applicable directly for the hybrid cloud and mobile cloud because the service architectures are different. Hence, multi-criteria optimization can offer the best possible solution in resource management, where multiple resource management metrics will be optimized simultaneously. Furthermore, multi-criteria-based optimization may bring new challenges if the considered metrics conflict due to the inter-dependency and optimization of one metric hampers the performance of the other metric, for example, energy-efficiency and network load, violations in SLA and maximization in profit, energy-efficiency and violations in SLA, etc. Thus, the objectives of resource management are to provide resource allocation and release in an energy-efficient manner, without SLA violation, increasing profit, balancing load, in public, private, and hybrid cloud as well as mobile cloud environment.

What Are the Different Types of Resource Management Methods?

According to the challenges and metrics discussed above, the resource management methods are categorized into the following seven categories in [3]:

- Energy-aware
- SLA-aware
- Market-based
- Load-balanced
- Network load-aware
- Hybrid cloud
- Mobile cloud computing (MCC).

In Sect. 3, we will discuss these seven types of methods for resource management.

In this chapter, various categories of cloud resources and resource management techniques are discussed. The rest of the article is organized as follows. Section 2 discusses different types of cloud resources. Section 3 discusses the categories of resource management techniques for cloud computing. Section 4 discusses the resource management in edge and fog computing. Section 5 highlights the challenges in resource management. Finally, Sect. 6 concludes the article.

2 Categories of Cloud Resources

The various categories of cloud resources are discussed briefly in this section.

2.1 Fast Computation Utility

The resources that offer *fast computation utility* in cloud computing environment come under this category. It includes efficient algorithms, size of memory, processing ability, etc. Fast computation utility offers computation as a service.

2.2 Storage Utility

The *storage utility* offers the facility of storing data at a remote place. A lot of hard drives, flash drives, database servers, etc. remove the barrier of limited storage capacity as in case of storing data in a local storage device. Storage utility offers storage as a service.

2.3 Communication Utility

It comprises the physical and logical resources in terms of hosts, sensors, communication link, intermediate nodes, protocols, bandwidth, delay, etc. This is also referred as Network utility that offers network as a service. Without communication utility, the storage utility and fast computation utility cannot be accessed. High-speed Internet connectivity is usually required in cloud service provisioning as delay and bandwidth are two vital parameters.

2.4 Energy/Power Utility

Along with delay and bandwidth, energy has also become a vital parameter in service provisioning. Low-power, i.e., energy-efficient approaches for cloud computing are on high demand. As a large number of data servers are used, the power consumption is very high in cloud computing. Thus, UPS and cooling devices are at the center of these resources, and they can be thought as the secondary resources.

2.5 Security Utility

The consumers require secure, highly reliable, trustworthy, and safe service provisioning. As the computation and storage take place remotely, security is a crucial issue of cloud computing. Hence, proper security mechanisms need to be applied in cloud computing.

3 Classification of Cloud Resource Management Methods

Resource management is a vital aspect of cloud computing for providing better performance and efficient utilization of underlying hardware. This section will discuss different types of resource management methods.

3.1 Energy-Aware Resource Management

The high energy consumption can result in the power expense enhancement of the service providers as well as increases the CO_2 emission. Therefore, energy efficiency is a significant issue in cloud computing. To deal with this issue, workload consolidation is important [3]. The energy consumption can be reduced through workload consolidation on a smaller number of servers. The VMs on the servers with less workload can be migrated to the servers with higher workload, and the idle servers can be turned off. Based on the modified best fit decreasing [3, 5], the VMs can be sorted in the descending order according to the requirements of CPU [3]. Thereafter, according to the power model, all VMs are allocated to the hosts. The power model allocates VM to the server with minimal change in energy consumption. To avoid SLA violations and keep the server usage within range, dynamic threshold values are used [3]. Along with energy efficiency, minimization of response time and maximization of availability are also vital [3, 6]. Non-linear optimization of resources can be performed against different timescales [3]. Using server partitioning available resources can be divided into multiple VMs. When a task is received, it can be placed on a particular group of servers based on the class. By predicting workload, resource allocation to running applications can be performed dynamically [3]. A central controller allocates and manages the resources. To minimize the overhead in decision-making and consequent energy consumption due to server power up, shutdown, and VM migration, the resource management decisions can be performed on hourly basis [3, 7].

3.2 SLA-Aware Resource Management

The SLA signed between the service provider and consumer contains information about the required service level, price, and penalty clause imposed if agreement is violated. Hence, SLA-aware resource management is significant. To manage fluctuating workload and ensure SLA, capacity allocation algorithms can play an important role [3, 8]. A workload predictor can be used for predicting future workload requirements [3, 8]. While the VMs communicate, the response time is kept lower to avoid SLA violations [3, 8]. Minimizing the probability of under or over resource provisioning is another vital aspect. In [3, 9], the algorithm uses various agents to provide and terminate resources. A prediction module has been also discussed to forecast the future service requirements [3, 9]. In [3, 10], task-oriented resource allocation has been discussed, where analytical hierarchy method and pair-wise comparison matrix method have been used for ranking resource allocation process. To handle the resources during their lifecycle, an SLA-aware platform has been discussed in [3, 11]. The SLA model is able to manage flexible requirements of multiple users and to deal with higher-level metrics [3, 11]. To minimize the SLA violations through elastic configuration of resources, a framework has been combined [3, 11].

3.3 Market-Based Resource Management

Cloud computing is a market-oriented paradigm. Thus, the service providers have the objective of increasing their profit. In [3, 12], auction-based resource allocation has been discussed, where the user bids for the advertised resources and if he/she wins then only resources are assigned. In [3, 13], multi-layer resource management has been discussed, where the SaaS provider provides service to the consumers only if they use of the services of IaaS providers. The SaaS provider uses resources of the IaaS provider and provides to the consumer. The IaaS provider uses optimal resource allocation method for enhancing own revenue. In order to maximize the revenue by minimizing the energy consumption but at the same time meeting the SLA, a model has been discussed in [3, 14], where each server contains a module for voltage/frequency scaling in a dynamic manner. To deal with load balancing, a hybrid optimization method has been also discussed. For cloud computing gold SLA and bronze SLA models have been discussed in [3, 15]. The gold SLA model considers average response time, reward value of each service request, maximum arrival rate of client request, and a penalty if there is a miss in average request response. The bronze SLA model is determined by the highest rate at which requests arrive and a utility function that calculates the profit for each request based on the response time.

3.4 Load-Balanced Resource Management

Load balancing is one of the important features in a computing environment. Load balancing usually refers to the process of workload sharing among multiple resources. In [3, 16], a dynamic load balancing approach has been proposed, where the workload is shifted from an overloaded server to a server that is less utilized. If two machines are less utilized, the workload of one machine is shifted to the other, and then switched off. In [3, 17], a load balancing method based on artificial neural network has been discussed. This method has used backpropagation for equal load distribution among all servers. Each user's demand is predicted and resource allocation takes place accordingly. However, at any given time, the active servers depend on the users' need at that particular time. Thus, active servers are minimized and energy consumption is reduced. In [3, 18], for resource management, max–min and min-min algorithms have been proposed. The completion times of all tasks on the different servers are determined and task with minimal completion time is chosen and allocated to the respective server. The task is then deleted from the list of unassigned tasks, and the same process is followed for rest of tasks. This process is continued until the list is empty. In max–min algorithm, the task with maximum completion time is chosen, and the similar process is followed. In [3, 19], a model has been discussed, where multi-dimensional resource set of each VM is considered and agents monitor the resources. For load balancing and respective resource allocation, game-theory-based methods have been also discussed.

3.5 Network Load-Aware Resource Management

In cloud computing, the consumer uses the service via Internet. Hence, the network load is an important factor. The network load refers to the traffic amount passing through the network at some specific time. In cloud computing, the resource managers share information, VM placement and migration take place, inter-VM communication occurs, hence, network load becomes vital. High network load degrades the performance because of the waiting time for VM placement and delay in inter-VM communication. Thus, minimization of traffic amount is significant and network load-aware resource management is required. In [3, 20], adaptive resource allocation has been discussed, where the distance between the user and the data center, and the workload of the data center, have been considered. In [3, 21], low perturbation bin packing algorithm has been discussed, and the list of servers has been sorted in descending way based on the energy consumption. Thereafter, a set or all of the VMs of the server at the top of the list are migrated to a server that has the lowest energy consumption. In [22], the clusters of VMs have been placed on the cluster of servers, where each virtual cluster offers a specific service type only, and the entire virtual

cluster is responsible to maintain the agreed QoS. For VM consolidation a genetic algorithm has been also proposed. In [23], based on network latency, the grouping of nodes is performed. The application scheduling is also performed based on network latency.

3.6 Hybrid Cloud Resource Management

The hybrid cloud is usually the integration of public and private clouds. In this case, the decision regarding the resource use of public cloud is significant. In [3, 23], rule-based resource manager has been discussed. The user requests are of two types: critical tasks with higher priority and secondary tasks with low priority. For securing the critical tasks or data, private cloud is selected for hosting them. On the other hand, the secondary tasks or data can utilize public as well as private cloud resources. However, the public cloud is only used if the private cloud is fully exhausted. The requests for VM provisioning can be categorized as low, medium, and high priority classes [3]. For class with high priority, new VMs are created, and among these new VMs, some are placed to the private cloud, and some are placed to the public cloud [3]. For medium priority class, user can request for two VMs, and both are placed to the private cloud [3]. For low priority class, one VM can be requested, and the VM is placed to the private cloud [3]. The incoming bag-of-task applications can be scheduled on the resources of private as well as public clouds using fully polynomial-time approximation scheme [3, 24].

3.7 MCC Resource Management

With the huge increase in the number of smartphone users, MCC has become a significant research domain. The traditional MCC follows an agent-client architecture, where the mobile devices use the resources available on the cloud. Nevertheless, mobile devices can share the resources in cooperation-based architecture as they have ample resources. MCC can be considered as an integration of various domains, where each domain contains cloud resources [3, 25]. To handle the cloud resources and provide continuous service to the users, the load can be shared among the domains. For improving customer satisfaction and reduce the number of service rejections, service request decision-making can be performed using semi-Markov decision process [3, 25]. The game theory can be used for minimizing overall power consumption, where all mobile devices act as players and migrate workload on one of the available servers for minimizing overall energy consumption [3, 26]. To handle the resources in MCC environment, nature-inspired chemical computing model has been used [27]. A middleware can be designed to formulate energy, bandwidth, and cloud resources in an MCC environment [3, 28].

4 Mathematical Model of Performance Evaluation Parameters

The performance evaluation involves considering and quantitatively defining the following parameters [3].

- Throughput
- Network overhead
- VM migration time
- Number of VM migrations
- Resource utilization
- Energy consumption
- Revenue and profit
- SLA violation.

4.1 Throughput

The throughput in cloud computing refers to the number of tasks finished in certain time period. The throughput (Thr) is mathematically determined as follows.

$$Thr = T_{tot} - T_{rem} \tag{1}$$

where T_{tot} and T_{rem} represent total number of received tasks and number of remaining, i.e., ongoing tasks, respectively.

4.2 Network Overhead

Network overhead is related to the network load described in Sect. 3. The network load ($Nload$) is mathematically determined as follows [3, 21].

$$Nload(P, t_1, t_2) = \sum_{i=1}^{V} \sum_{j=1}^{V} \sum_{a=1}^{S} \sum_{b=1}^{S} C_{ij}^{t_1,t_2} f_{ia} f_{jb} d_{ab} \tag{2}$$

where $C_{ij}^{t_1,t_2}$ presents a $(V \times V)$ matrix representing the data amount exchanged between VMs V_j and V_i in time interval $(t_1 - t_2)$, f_{ia} indicates whether V_i is hosted on server S_a (if hosted the value is 1, otherwise the value is 0), f_{jb} indicates whether V_j is hosted on server S_b (if hosted the value is 1, otherwise the value is 0), d_{ab} is the cost to exchange one abstract data unit between S_a and S_b, P denotes VM placement, V denotes the number of VMs, and S denotes the number of servers.

4.3 VM Migration Time

The migration time ($Time_k$) for a VM, V_k, is determined as follows [3, 29].

$$Time_k = \frac{M_k}{B_k} \tag{3}$$

where M_k and B_k represents the memory amount used by V_k and the available bandwidth respectively.

4.4 Number of VM Migrations

The number of VM migrations (N_{mig}) in a given time interval ($t_1 - t_2$) is determined as follows [3].

$$N_{mig}(P, t_1, t_2) = \sum_{a=1}^{S} \int_{t_1}^{t_2} Mig_a(P) \tag{4}$$

where P denotes the present VM placement, $Mig_a(P)$ represents the number of migrations of server S_a in time interval ($t_1 - t_2$) in case of placement P.

4.5 Resource Utilization

The utilization of S_a at time t can be mathematically represented by the following equation [3, 21].

$$U_a(P, t) = \sum_{j=1}^{V} f_{aj} * \frac{R_{CPU_j}(t)}{CPU_a} \tag{5}$$

where P denotes the present VM placement, f_{ja} indicates whether V_j is hosted on server S_a (if hosted the value is 1, otherwise the value is 0), CPU_a denotes computational capacity of S_a, $R_{CPU\,j}(t)$ represents the CPU capacity that V_j requires at time t, and V denotes the number of VMs.

4.6 Energy Consumption

The power consumed by S_a at time t is determined as follows [3, 5, 21].

$$Pow_a(P,t) = 0.7Pow^a_{max} + 0.3Pow^a_{max} * U_a(P,t) \tag{6}$$

where Pow^a_{max} denotes the power consumption of server S_a in case of full utilization and denotes the utilization of S_a at time t. The total energy consumed by all servers in time interval $(t_1 - t_2)$ is calculated as follows [3, 5, 21].

$$En(P,t_1,t_2) = \sum_{a=1}^{S} \int_{t_1}^{t_2} Pow_a(P,t) \tag{7}$$

4.7 Revenue and Profit

The profit (Profit) is calculated based on total revenue (Revenue) and total expenditure (Expenditure) as follows [3].

$$\text{Profit} = \text{Revenue} - \text{Expenditure} \tag{8}$$

4.8 SLA Violation

The SLA violation can be calculated as follows [3, 29].

$$\text{violation}_{\text{sla}} = T_{\text{slavah}} \cdot \text{Perdeg}_{\text{mig}} \tag{9}$$

where T_{slavah} denotes each active host's SLA violation time, and $\text{Perdeg}_{\text{mig}}$ denotes the performance degradation for migrations.

5 Resource Management for Edge and Fog Computing

Nowadays, edge and fog computing have become promising technologies in computing. The use of remote cloud suffers from various drawbacks, such as high latency, high energy consumption, high network overhead, security, etc. As a result,

the QoS degrades. To overcome the difficulties, edge computing and fog computing have arrived. Edge computing involves the relocation of resources to the network edge, while fog computing involves the involvement of intermediate devices in computation instead of relying solely on the cloud for computation [30–32]. The resource management in edge computing and fog computing is an emerging area of research. Along with edge and fog computing, Internet of Things (IoT) has also gained a lot of interest of the researchers [33–36]. The edge and fog computing with IoT are leading to provide smart solutions to real-life scenarios [31, 37–40].

5.1 Resource Management in Edge Computing

The edge devices have limited resources, and edge resources are heterogeneous [41]. The resource management architectures used for edge computing are categorized depending on the data flow, control, and tenancy [41].

The *data flow architectures* depend upon the direction in which workloads and data travel within the computer ecosystem. In this scenario, the workloads can be shifted from the user device to the edge device, or from the cloud to the edge device.

The *control architectures* are contingent upon the manner in which the resources are regulated inside the computing ecosystem. In this case, a single controller or a central algorithm can be used to manage the edge nodes, or a distributed method can be used.

The tenancy architectures rely on the level of support provided for hosting different entities inside the ecosystem. The edge node in this scenario can host one application or several.

5.2 Resource Management Issues of Fog Computing

Fog computing offers the facility of processing and storing data locally rather putting the entire overhead on the cloud. The fog computing offers storage, networking, and computing resources. The issues of resource management in case of fog computing are divided into the following categories [42]:

- Application placement
- Resource scheduling
- Resource allocation
- Resource provisioning
- Task offloading
- Load balancing.

5.2.1 Application Placement

The application placement schemes are categorized as centralized, decentralized, and hierarchical [42]. In centralized schemes, the broker requires information from all entities for global optimization in decision. In decentralized schemes, the broker contains some portion of the information and it is applicable for the environments, which have smaller number of components. To offer the benefits of both the centralized and decentralized schemes, the hierarchical approaches focus on semi-global and local managers and both the managers work together.

5.2.2 Resource Scheduling

The resource scheduling approaches are categorized into three distinct groups: static, dynamic, and hybrid [42]. In case of static scheduling, the tasks reach the fog nodes concurrently, and before submission of the tasks the decisions regarding scheduling are made. In case of dynamic scheduling, the tasks' arrival times are unknown, and the scheduling takes place once the tasks are submitted. In case of hybrid scheduling, various scheduling criteria are merged to cover various categories of allocation, for example, batch jobs and workflows.

5.2.3 Resource Allocation

The methods of resource allocation are divided into two categories: auction-based methods and optimization methods [42]. The resource allocation methods that utilize auctions employ market-oriented pricing to manage the demand and supply of fog nodes. These methods involve putting fog nodes up for bidding and subsequently assigning them to the bidder with the highest offer. In case of optimization methods, the resource allocation is considered as a double-matching. Consequently, there is a coupling between fog nodes and cloud servers for IoT users, and a coupling between IoT users and fog nodes for cloud servers.

5.2.4 Resource Provisioning

The resource provisioning schemes are divided into three categories: reactive, proactive, and hybrid [42]. In case of reactive policy, no prediction is performed and response is provided to the present system status. In case of proactive policy, future demands of IoT applications are predicted for updating resource provisioning with enough anticipation. In case of hybrid policy, the reactive policy is used for fog node provisioning and the proactive policy is used for fog node releasing.

5.2.5 Task Offloading

The process of transferring computation-intensive tasks from low-resource devices to resource-rich devices is referred as task offloading. On the basis of the number of offloading destinations, the task offloading methods are categorized as single-type and multiple-type offloading methods [42]. For single-type offloading, the computing tasks are transferred to a single fog node and processed sequentially. In case of multiple-type offloading, the computational tasks are offloaded using multiple fog nodes and parallel processing takes place.

5.2.6 Load Balancing

The load balancing methods are categorized as centralized, decentralized, and hybrid [42]. In case of a centralized approach, a central node is responsible for the load balancing. Under a decentralized approach, the system's nodes are separated into clusters, with each cluster utilizing central nodes to distribute the system's workload evenly. The hybrid approaches provide a compromise between the advantages of both centralized and decentralized approaches for load balancing.

6 Resource Management Systems and Simulation Tools

In this section, we briefly mention the resource management software and tools used for experimental analysis and simulation.

6.1 Resource Management Systems in Practice

There are various cloud platforms such as Google Cloud platform [43], Microsoft Azure [44], Amazon Web Services [45], etc. Aneka [46, 47] is another well-known application development platform for cloud computing. Hadoop [48, 49] is popular for big data analysis in cloud platform. Kubernetes [50] is well-known for container orchestration. The summary of the characteristics of the existing cloud resource management systems and platforms is presented in Table 1.

6.2 Simulators for Resource Management

CloudSim [51] is a well-known simulator used for simulating the cloud computing environment. The iFogSim [52] is an extended version of CloudSim that is used for fog computing. To support mobility, clustering, and microservice management

Table 1 The characteristics of cloud platforms

Name of the system	Characteristics
Google Cloud Platform (GCP)	GCP is a cloud computing service suite that offers data storage, analysis, computing, and management tools. GCP has resource manager to manage the resources hierarchically
Microsoft Azure	Microsoft Azure offers a set of cloud services such as data storage, database hosting, artificial intelligence, and Internet of Things-based services. It allows running virtualized computers to manage customized software solutions
Amazon AWS	AWS provides a range of cloud services encompassing storage, processing, data analysis, IoT, and more. AWS has an AWS resource access manager
Aneka	Aneka is a popular cloud computing platform that offers physical and virtualized resources, which are connected using a network. Every resource possesses an instance of Aneka container, which serves as the operational environment for executing distributed applications
Hadoop	Apache Hadoop offers a collection of open-source software for distributed storage and big data analysis using MapReduce. For resource management and scheduling, Hadoop's computation platform YARN is popular
Kubernetes	Kubernetes is an open-source technology used for coordinating and managing containers, enabling automated deployment, scaling, and administration of software

Table 2 The characteristics of Simulation tools

Name of the tool	Characteristics
CloudSim	This toolkit is designed to simulate extendable clouds. CloudSim allows for the expansion and specification of rules inside the software stack components
iFogSim	iFogSim is a software toolkit used for simulating and modeling IoT and fog computing environments
iFogSim2	iFogSim2 simulator is an extension of iFogSim. iFogSim2 addresses the service migration for various mobility models for the Internet of Things devices, distributes cluster information among the edge and fog nodes, and provides microservice management
EdgeSimPy	EdgeSimPy is a Python-based simulation framework used for modeling and to evaluate policies of resource management in edge computing scenarios

iFogSim2 [53] is used. For edge computing, EdgeSim [54] is used. The characteristics of the simulators are summarized in Table 2.

7 Research Challenges

This section highlights the remaining research issues in resource management in cloud computing.

7.1 Consumer-Based Service Management

User satisfaction is highly significant for cloud computing, which is a market-oriented paradigm. To enhance the customer satisfaction level, user-centric objectives can be considered along with the customer service requirements, customer's profile, etc. [3]. The communication with consumers and their feedback can also help to improve the user experience. Reliability and trustworthiness are also play important roles in customer satisfaction.

7.2 Autonomic Resource Management

The resource demands in a cloud computing environment change over time [3]. The prediction of workload in elastic data centers and appropriate resource provisioning are also vital in cloud computing. Thus, autonomic resource management methods are required to develop.

7.3 Resource Information Management

Resource information collection from different server sets increases the network overhead, and the analysis of the information increases the processing overhead of the central manager [3]. Distributed cluster managers can be used for collecting and analyzing resource information and taking action accordingly in respective clusters without affecting the functions of other clusters. A central manager can also be used to select the host cluster on the basis of the information given by the cluster managers.

7.4 Heterogeneous Resources

The load balancing methods should consider the architecture of the computing server [3]. In case of VM migration also information of the destination is highly important, otherwise, the performance may degrade, for example, VM migration to a server with less cache will degrade the performance.

7.5 Sharing of Network Resources

Bandwidth management and network virtualization are significant in a cloud computing environment [3]. For sharing network resources but isolating workload

of different users, solutions are required. Workload isolation is significant as the variation in load of one user can hamper other users' services.

7.6 Security

Within a cloud computing setting, the server resources are distributed among numerous users [3]. Thus, there is a possibility of data breach. Hence, secure resource sharing is required.

7.7 Large-Scale Cloud Management

The number of cloud service users is growing rapidly, and accordingly, the service requests and amount of data are also increasing [3]. The number of communications can be increased due to the large and disperse operations, and consequently, the network load can increase. Smart resource allocation methods for interdependent sub-task placement on same cluster servers are required. Furthermore, the latency as well as network load is required to be minimized.

7.8 Computational Risk Analysis and Management

The cloud computing provides computing and storage as services. Nevertheless, there are various risks [3], for example, the difficulties due to network load, system failure, resource manager load, inadequate resources, may occur, which can lead to SLA violation. Hence, risk analysis and management by providing solutions to these problems need to be studied.

7.9 Multi-parametric Performance Evaluation

In a real-time environment, the user demands for resources vary and any user can generate any type of service request any time. Furthermore, the resources are distributed. Thus, the performance of the resource management methods needs to be evaluated based on various parameters [3], such as various sizes of VMs, changing service requirements, types of resources, number of resources, different workloads, etc.

7.10 Service Benchmarking

In cloud computing, various consumers have various service requirements [3]. The requirements of various types of services are different, for example, some services are computation-intensive, some are data-intensive, some are sensitive and need security, etc. Thus, realistic benchmark for evaluating various types of resource management methods in such heterogeneous service requirements is required.

7.11 Robustness

The failure of network link or node may result in communication failure between the servers and VMs [3]. As a result, VM migration is affected and accordingly the performance degrades. If server failure occurs, the executing tasks as well as the result are lost, which results in service failure. Hence, the system should be robust enough to deal with these issues.

8 Summary

This chapter discussed the resource types, resource management issues, approaches, and challenges in a cloud computing environment. The resource management for edge and fog computing is also discussed. Finally, the future research challenges of resource management are highlighted in this chapter.

References

1. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Futur. Gener. Comput. Syst.. Gener. Comput. Syst. **25**(6), 599–616 (2009)
2. Parikh, S.M., Patel, N.M., Prajapati, H.B.: Resource Management in Cloud Computing: Classification and Taxonomy (2017). arXiv:1703.00374
3. Mustafa, S., Nazir, B., Hayat, A., Madani, S.A.: Resource management in cloud computing: taxonomy, prospects, and challenges. Comput. Electr. Eng.. Electr. Eng. **47**, 186–203 (2015)
4. Beloglazov, A., Abawajy, J., Buyya, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Futur. Gener. Comput. Syst. Gener. Comput. Syst. **28**(5), 755–768 (2012)
5. Sobers Smiles David, G., Ramkumar, K., Shanmugavadivu, P., Eliahim Jeevaraj, P.S.: Introduction to cloud resource management. In: Choudhury, T., Dewangan, B.K., Tomar, R., Singh, B.K., Toe, T.T., Nhu, N.G. (eds.) Autonomic Computing in Cloud Resource Management in Industry 4.0. EAI/Springer Innovations in Communication and Computing. Springer, Cham (2021)

6. Addis, B., Ardagna, D., Panicucci, B., Squillante, M.S., Zhang, L.: A hierarchical approach for the resource management of very large cloud platforms. IEEE Trans. Dependable Secure Comput.Comput. **10**(5), 253–272 (2013)
7. Ardagna, D., Panicucci, B., Trubian, M., Zhang, L.: Energy-aware autonomic resource allocation in multitier virtualized environments. IEEE Trans. Serv. Comput.Comput. **5**(1), 2–19 (2010)
8. Ardagna, D., Casolari, S., Colajanni, M., Panicucci, B.: Dual time-scale distributed capacity allocation and load redirect algorithms for cloud systems. J. Parallel Distrib. Comput. **72**(6), 796–808 (2012)
9. Wei, Y., Blake, M.B., Saleh, I.: Adaptive resource management for service workflows in cloud environments. In: 2013 IEEE International Symposium on Parallel and Distributed Processing, Workshops and Phd Forum, pp. 2147–2156. IEEE (2013)
10. Ergu, D., Kou, G., Peng, Y., Shi, Y., Shi, Y.: The analytic hierarchy process: task scheduling and resource allocation in cloud computing environment. J. Supercomput.Supercomput. **64**, 835–848 (2013)
11. García, A.G., Espert, I.B., García, V.H.: SLA-driven dynamic cloud resource management. Futur. Gener. Comput. Syst.. Gener. Comput. Syst. **31**, 1–11 (2014)
12. Zaman, S., Grosu, D.: A combinatorial auction-based mechanism for dynamic VM provisioning and allocation in clouds. IEEE Trans. Cloud Comput. **1**(2), 129–141 (2013)
13. Chunlin, L., Layuan, L.: Multi-layer resource management in cloud computing. J. Netw. Syst. Manag.Netw. Syst. Manag. **22**, 100–120 (2014)
14. Ali, S., Jing, S.Y., Kun, S.: Profit-aware DVFS enabled resource management of IaaS cloud. Int. J. Comput. Sci. Issues (IJCSI) **10**(2 Part 2), 237 (2013)
15. Goudarzi, H., Pedram, M.: Profit-maximizing resource allocation for multitier cloud computing systems under service level agreements. In: Large Scale Network-Centric Distributed Systems, pp. 291–318 (2013)
16. Ban, Y., Chen, H., Wang, Z.: Ealarm: Enhanced autonomic load-aware resource management for p2p key-value storage in cloud. In: 2013 IEEE Seventh International Symposium on Service-Oriented System Engineering, pp. 150–155. IEEE (2013)
17. Al Sallami, N.M., Al Alousi, S.A.: Load balancing with neural network. IJACSA Int. J. Adv. Comput. Sci. Appl. **4**(10) (2013)
18. Kokilavani, T., Amalarethinam, D.G.: Load balanced min-min algorithm for static meta-task scheduling in grid computing. Int. J. Comput. Appl.Comput. Appl. **20**(2), 43–49 (2011)
19. Ye, D., Chen, J.: Non-cooperative games on multidimensional resource allocation. Futur. Gener. Comput. Syst.. Gener. Comput. Syst. **29**(6), 1345–1352 (2013)
20. Jung, G., Sim, K.M.: Agent-based adaptive resource allocation on the cloud computing environment. In: 2011 40th International Conference on Parallel Processing Workshops, pp. 345–351. IEEE (2011)
21. Tziritas, N., Xu, C.Z., Loukopoulos, T., Khan, S.U., Yu, Z.: Application-aware workload consolidation to minimize both energy consumption and network load in cloud environments. In: 2013 42nd International Conference on Parallel Processing, pp. 449–457. IEEE (2013)
22. He, L., Zou, D., Zhang, Z., Chen, C., Jin, H., Jarvis, S.A.: Developing resource consolidation frameworks for moldable virtual machines in clouds. Futur. Gener. Comput. Syst.. Gener. Comput. Syst. **32**, 69–81 (2014)
23. Malik, S., Huet, F., Caromel, D.: Latency based group discovery algorithm for network aware cloud scheduling. Futur. Gener. Comput. Syst.. Gener. Comput. Syst. **31**, 28–39 (2014)
24. Farahabady, M.R.H., Lee, Y.C., Zomaya, A.Y.: Pareto-optimal cloud bursting. IEEE Trans. Parallel Distrib. Syst.Distrib. Syst. **25**(10), 2670–2682 (2013)
25. Liang, H., Cai, L.X., Huang, D., Shen, X., Peng, D.: An SMDP-based service model for interdomain resource allocation in mobile cloud networks. IEEE Trans. Veh. Technol. **61**(5), 2222–2232 (2012)
26. Ge, Y., Zhang, Y., Qiu, Q., Lu, Y.H.: A game theoretic resource allocation for overall energy minimization in mobile cloud computing system. In: Proceedings of the 2012 ACM/IEEE International Symposium on Low Power Electronics and Design, pp. 279–284 (2012)

27. Ikram, A., Anjum, A., Bessis, N.: A cloud resource management model for the creation and orchestration of social communities. Simul. Model. Pract. TheoryPract. Theory **50**, 130–150 (2015)
28. O'Sullivan, M.J., Grigoras, D.: Integrating mobile and cloud resources management using the cloud personal assistant. Simul. Model. Pract. TheoryPract. Theory **50**, 20–41 (2015)
29. Beloglazov, A., Buyya, R.: Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concurrency Comput. Pract. Experience **24**(13), 1397–1420 (2012)
30. Mukherjee, A., De, D., Ghosh, S.K., Buyya, R.: Introduction to Mobile Edge Computing, pp. 3–19. Springer International Publishing (2021)
31. Mukherjee, A., Deb, P., De, D., Buyya, R.: C2OF2N: a low power cooperative code offloading method for femtolet-based fog network. J. Supercomput.Supercomput. **74**, 2412–2448 (2018)
32. Mukherjee, A., Ghosh, S., De, D., Ghosh, S.K.: Mcg: mobility-aware computation offloading in edge using weighted majority game. IEEE Trans. Netw. Sci. Eng. **9**(6), 4310–4321 (2022)
33. Shah, S.H., Yaqoob, I.: A survey: Internet of Things (IOT) technologies, applications and challenges. IEEE Smart Energy Grid Eng. (SEGE) 381–385 (2016)
34. Mukherjee, A., Deb, P., De, D., Buyya, R.: IoT-F2N: an energy-efficient architectural model for IoT using Femtolet-based fog network. J. Supercomput.Supercomput. **75**, 7125–7146 (2019)
35. Shafique, K., Khawaja, B.A., Sabir, F., Qazi, S., Mustaqim, M.: Internet of things (IoT) for next-generation smart systems: a review of current challenges, future trends and prospects for emerging 5G-IoT scenarios. IEEE Access **8**, 23022–23040 (2020)
36. Koohang, A., Sargent, C.S., Nord, J.H., Paliszkiewicz, J.: Internet of Things (IoT): from awareness to continued use. Int. J. Inf. Manag. **62**, 102442 (2022)
37. Mukherjee, A., De, D., Buyya, R: E2R-F2N: Energy-efficient retailing using a femtolet-based fog network. Softw. Pract. Experience **49**(3), 498–523 (2019)
38. Mukherjee, A., De, D., Ghosh, S.K.: FogIoHT: a weighted majority game theory based energy-efficient delay-sensitive fog network for internet of health things. Internet Things **11**, 100181 (2020)
39. Yudidharma, A., Nathaniel, N., Gimli, T.N., Achmad, S., Kurniawan, A.: A systematic literature review: messaging protocols and electronic platforms used in the internet of things for the purpose of building smart homes. Procedia Comput. Sci. **216**, 194–203 (2023)
40. Bera, S., Dey, T., Mukherjee, A., Buyya, R.: E-CropReco: a dew-edge-based multi-parametric crop recommendation framework for internet of agricultural things. J. Supercomput. 1–35 (2023)
41. Hong, C.H., Varghese, B.: Resource management in fog/edge computing: a survey on architectures, infrastructure, and algorithms. ACM Comput. Surv. (CSUR) **52**(5), 1–37 (2019)
42. Ghobaei-Arani, M., Souri, A., Rahmanian, A.A.: Resource management approaches in fog computing: a comprehensive review. J. Grid Comput. **18**(1), 1–42 (2020)
43. Bisong, E., Bisong, E.: An overview of google cloud platform services. Build. Mach. Learn. Deep Learn. Models Google Cloud Platform: Compr. Guide Beginners 7–10 (2019)
44. Collier, M., Shahan, R.: Microsoft Azure Essentials-Fundamentals of Azure. Microsoft Press (2015)
45. Mathew, S., Varia, J.: Overview of amazon web services. Amazon Whitepapers **105**, 1–22 (2014)
46. Chu, X., Nadiminti, K., Jin, C., Venugopal, S., Buyya, R.: Aneka: Next-generation enterprise grid platform for e-science and e-business applications. In: Third IEEE International Conference on e-Science and Grid Computing (e-Science 2007), pp. 151–159. IEEE (2007)
47. Toosi, A.N., Sinnott, R.O., Buyya, R.: Resource provisioning for data-intensive applications with deadline constraints on hybrid clouds using Aneka. Futur. Gener. Comput. Syst.. Gener. Comput. Syst. **79**, 765–775 (2018)
48. Nandimath, J., Banerjee, E., Patil, A., Kakade, P., Vaidya, S., Chaturvedi, D.: Big data analysis using Apache Hadoop. In: 2013 IEEE 14th International Conference on Information Reuse and Integration (IRI), pp. 700–703. IEEE (2013)

49. Vavilapalli, V.K., Murthy, A.C., Douglas, C., Agarwal, S., Konar, M., Evans, R., et al.: Apache hadoop yarn: Yet another resource negotiator. In: Proceedings of the 4th Annual Symposium on Cloud Computing, pp. 1–16 (2013)
50. Chang, C.C., Yang, S.R., Yeh, E.H., Lin, P., Jeng, J.Y.: A kubernetes-based monitoring platform for dynamic cloud resource provisioning. In: GLOBECOM 2017–2017 IEEE Global Communications Conference, pp. 1–6. IEEE (2017)
51. Calheiros, R.N., Ranjan, R., Beloglazov, A., De Rose, C.A., Buyya, R.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Softw. Pract. Experience **41**(1), 23–50 (2011)
52. Gupta, H., Vahid Dastjerdi, A., Ghosh, S.K., Buyya, R.: iFogSim: a toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments. Softw. Pract. Experience **47**(9), 1275–1296 (2017)
53. Mahmud, R., Pallewatta, S., Goudarzi, M., Buyya, R.: Ifogsim2: an extended ifogsim simulator for mobility, clustering, and microservice management in edge and fog computing environments. J. Syst. Softw.Softw. **190**, 111351 (2022)
54. Souza, P.S., Ferreto, T., Calheiros, R.N.: EdgeSimPy: Python-based modeling and simulation of edge computing resource management policies. Futur. Gener. Comput. Syst. (2023)