ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 13 (2023) No. 5, pp. 107-126
DOI:10.33168/JSMS.2023.0507

A Review on Quality of Service Monitoring, Violation and Remediation for the Cloud

Hassan Mahmood Khan, Fang-Fang Chua, Timothy Tzen Vun Yap

Faculty of Computing and Informatics, Multimedia University, Cyberjaya 63100, Malaysia

hasmkh@gmail.com, ffchua@mmu.edu.my, timothy@mmu.edu.my

Abstract. Cloud computing has emerged as a dominant paradigm for delivering scalable and on-demand computing resources to users worldwide. However, ensuring Quality of Service (QoS) in cloud environments remains a critical challenge. This review paper comprehensively analyzes the state-of-the-art techniques, methodologies, and frameworks for QoS monitoring, violation detection, and remediation in cloud-based systems. This paper examines studies related to the concepts of QoS monitoring, violations, and remediation using resource allocation and scalability in cloud computing. It provides a taxonomy of QoS metrics, including availability, response time, and throughput, that are essential for evaluating and maintaining the performance of cloud services. The review further examines the existing approaches for cloud QoS monitoring, ranging from infrastructure to application-level monitoring. It discusses various monitoring tools and technologies employed to collect and analyze QoS data, including cloud type, license, operating systems, and supported languages. Additionally, the paper investigates the techniques for detecting QoS violations in the cloud environment. It explores machine learning algorithms and hybrid methods that leverage a combination of these techniques to identify QoS violations accurately. Furthermore, the paper explores the remediation strategies for QoS violations in the cloud. It presents proactive and reactive approaches to address QoS breaches, such as QoS-aware, energy-efficient, dynamic, multidimensional deadline-based, load balancing, cost-aware, workload, scalability, and adaptive resource allocation. It also discusses the challenges and validity threats associated with this research study. Throughout the review, the paper identifies the strengths and limitations of the existing approaches. It provides insights into future research directions in QoS monitoring, violation detection, and remediation for cloud-based systems. It emphasizes the need for standardized frameworks and benchmarks to evaluate and compare different QoS monitoring and remediation techniques. In conclusion, this review paper consolidates the current knowledge and advancements in QoS monitoring, violation detection, and remediation for cloud computing. It serves as a valuable resource for researchers seeking to enhance QoS assurance in cloud environments, ultimately improving the trust and performance of cloud services.

Keywords: Cloud Computing, Quality of Service, Cloud Monitoring, Service Level Agreement, Resource Management, Violation Detection, Violation Remediation

1. Introduction

Cloud computing is a new technology paradigm that facilitates users with scalable, convenient, and virtualized resources. Cloud infrastructure provides the hardware and software resources such as CPU, memory, storage, and specific application to the clients on a pay-per-use policy through the Internet Yakubu IZ, Musa ZA, Muhammed L, et al. (2020). According to the client's requirements, cloud resources can automatically be scaled up and down. Similarly, cloud computing has a few other advantages: easy to use, less operational cost, flexibility, efficiency, automatic updates, security, sustainability, and easily manageable. Cloud computing offers three service layers (Bokhari et al. 2016). The first layer is the Software as a Service (SaaS) which provides software and applications developed at the user end. SaaS provides facilities for a wide range of services, and customers gain advantages from its cost-effective and affordable long-term solutions via the Internet. The characteristics of the SaaS involved disaster recovery, scalable resources, autonomous access to key applications, and eradicating the need for application supervision (Kohlbrenner, 2021). The Platform as a Service (PaaS) is the second layer responsible for developing and deploying the software. A built-in environment is provided to the clients, such as server hardware and software, network environment, and operating systems. The Infrastructure as a Service (IaaS) layer is at the tip of the cloud computing pyramid. The vendor provides service users with storage, network servers, and other cloud resources on a pay-per-use policy. This way, the IaaS layer allows access to all hardware and other mandatory tools to build and run datacentres. Under a cloud computing environment, there are four types of cloud models, such as public cloud, private cloud, community cloud, and hybrid cloud, in terms of customer requirements.

Trust management in cloud computing is challenging and acts as a promise between service providers and customers. In simple words, trust is the degree to which customers are agreeable to relying on cloud service; that provider provides the recourse with specific qualities that are assured to be delivered by the cloud providers (Challagidad & Birje, 2017). Moreover, trust management is classified from two different perspectives: service provider that belongs to cloud providers and service requester that belongs to customer demands. Further, there are four ways in cloud management to establish trust between clients and vendors; the first is by using a set of policies to develop trust in cloud infrastructure. The second is the recommendation that works as a psychology theory and benefits from customers' knowledge of trusted companies. The third is reputation maintained based on the customer's feedback that can intensely change cloud service providers' status, positively or negatively. The fourth is prediction; when no prior information is available, prediction is a trust management technique used in many cloud environments to gain trust (Hayyolalam, Pourghebleh, & Pourhaji Kazem, 2020).

One of the main tasks of cloud computing is cloud monitoring. In cloud monitoring, the supervision of cloud resources involves investigating, managing, and evaluating. The uninterrupted monitoring of the cloud offers benefits to cloud customers and cloud service providers in terms of performance, adaptability, availability, and timeliness. Quality of Service (QoS), as guaranteed by the cloud providers, should involve different metrics with throughput, CPU and memory utilization, latency, and processing time in terms of performance. Additionally, resources allocated to the customers should come with promising QoS metrics as agreed in the signed contract, namely, the Service Level Agreement (SLA). In case of violation of these QoS parameters, the providers should be penalized in financial terms or other alternatives as remedial measures. There should be a mechanism for QoS violation remediation.

The Objective of this research is “to examines the existing approaches for cloud QoS monitoring, QoS Vioation Detection and Remediation Techniques in the cloud computing”. This study discusses various monitoring tools and technologies employed to collect and analyze QoS data, including cloud type, license etc. This work also investigates the techniques for detecting QoS violations in the cloud environment. It explores machine learning algorithms and hybrid methods that leverage a combination of these techniques to identify QoS violations.

This paper explores the remediation strategies for QoS violations in the cloud. It presents approaches to address QoS breaches, such as QoS-aware, energy-efficient, dynamic, multidimensional deadline-based, load balancing, cost-aware, workload, scalability, and adaptive resource allocation.

While a review paper primarily reviews existing knowledge and research in a specific domain, it also offers unique contributions to the field. It includes

I. Evaluation of Monitoring Approaches: The review paper examines various approaches for QoS monitoring in the cloud, ranging from infrastructure-level to application-level monitoring. It critically analyzes the strengths and limitations of different monitoring tools and technologies used for collecting and analyzing QoS data. This evaluation aids in understanding the trade-offs and challenges associated with different monitoring approaches.
II. Overview of Violation Detection Techniques: The paper explores the techniques for detecting QoS violations in the cloud environment. It discusses machine learning algorithms and hybrid methods that are employed for accurate identification of QoS breaches. This overview provides insights into the effectiveness and applicability of different violation detection techniques.
III. Examination of Remediation Strategies: The review investigates the remediation strategies for addressing QoS violations in the cloud. It presents proactive and reactive approaches, QoS-aware, energy-efficient, dynamic, multidimensional deadline-based, load balancing, cost-aware, workload, scalability, and adaptive resource allocation.
IV. Identification of Future Research Directions: The review paper identifies future research directions in QoS monitoring, violation detection, and remediation for cloud-based systems. It highlights the need for real-time monitoring, adaptive and intelligent remediation, standardized SLA frameworks, and advancements in multi-tenant and multi-cloud environments. These identified research directions serve as a roadmap for future and guide researchers in addressing emerging challenges.

Overall, the review paper's research contribution lies in consolidating existing knowledge, providing a comprehensive analysis, and identifying future research directions. It serves as a valuable resource for researchers, guiding them in enhancing QoS assurance in cloud environments and improving the trust and performance of cloud services. The rest of the paper is organized as follows: Sect. 2 defines the research methodology and questions; Sect. 3 gives the key findings; Sect. 4 presents an analysis and discussion of the literature findings; Sect. 5 illustrates the validity threat of this study; and Sect. 6 presents conclusions and future directions.

2. Research Methodology

A literature review was selected as the research approach for this study because it is one of the most successful ways of discovering, analyzing, interpreting, and comparing all available studies relevant to a specific subject. Such a method might result in detailed replies within a defined area. In this regard, we devised three research topics to address the significant challenges of cloud computing: QoS in cloud computing, QoS violation detection, and QoS violation remediation. We reviewed the most current writings in the areas specified. In particular, we were looking for methodologies, frameworks, prototypes, and commercial solutions that addressed the abovementioned challenges. ACM Digital Library, IEEE Xplore Digital Library, Elsevier ScienceDirect, and SpringerLink are the databases considered in this study. We also considered grey literature for the QoS monitoring tools. Although many studies have previously been published pertinent to cloud monitoring, relatively few address QoS violations and remediation. These studies each identify and categorize various activities conducted under the broad awning of cloud monitoring and resource allocation under multiple viewpoints. The

main goal of the research provided in this work is to present an updated, comprehensive analysis of the current efforts that incorporate analytical viewpoints from earlier studies and finally pinpoint the QoS violations and remedial goals that are yet an open research area.
The literature review research process is shown in Figure 1, which contains eight steps.

I. The first step of this research is defining the research questions, which provides the research directions.
II. The second step is the review of the research scope to determine the analysis dimensions of the survey.
III. The third step is keyword selection, based on steps one and two. It helps to obtain the relevant publication list; a generic search string is created. It contains QoS monitoring, violation, remediation, solutions, methods, and framework keywords. As QoS word is used in many research areas, we have used cloud computing keywords in the search string.

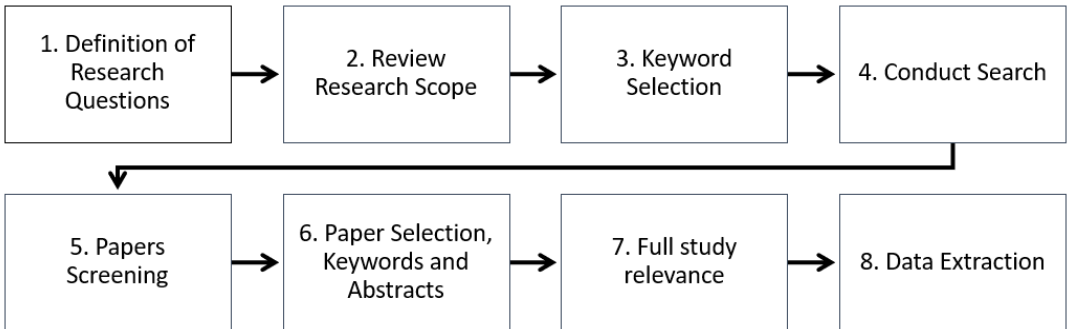

Fig.1: : The Research Methodology Process

IV. The fourth step is to search; the search string with duration (2016- 2022) has been applied to all four libraries (ACM, IEEE, Elsevier and Springer), as mentioned above.
V. In step 5, searched papers are screened for this study's suitability. Title, abstract, and keywords are considered for the screening at first.
VI. After that most relevant articles are selected in step six.
VII. Some studies were also eliminated during the entire study review due to the non-relevance of the research topic in step seven.
VIII. After choosing the most relevant studies, we extract the data from the selected studies in step eight.

Criteria for inclusion:

- Publications in the discipline of computer science
- Publications in the domain of computer science
- Publications specifically relevant to cloud QoS monitoring

Criteria for exclusion:

- Publications that are not in English.
- Publications with no full article available.
- Publications that are not accessible online.
- Publications that are identical to earlier publications

2.1.Research Questions

The main contributions of this paper are as follows.

RQ1: What are the QoS monitoring tools for cloud computing? What are QoS standards, and which are considerable QoS parameters?

The aim is to explore the QoS monitoring tools available for cloud computing, the current cloud QoS standards, and a list of important QoS parameters.

RQ2: How is QoS violation detection in cloud computing carried out?

The aim is to find contributions from existing literature on how QoS violation detection problem is solved.

RQ3: What is the state-of-the-art for QoS violation remediation in cloud computing?

In the case of QoS violation, the aim is to understand which methodologies are used and how QoS violation remediation is implemented in Cloud Computing.

The study analyses the state-of-the-art based on the research questions and presents the research issues and challenges of QoS violation detection and remediation for future guidelines.

3. Research Findings on Cloud Computing

This section describes the findings of the literature review. The following subsections will examine the results prompted by the three previously mentioned research questions.

3.1. Cloud QoS Monitoring

3.1.1. Cloud Monitoring Tools

Cloud monitoring involves tracking the various features related to Quality of Service (QoS) and incorporating them with the overall cloud management strategies. For instance, cloud monitoring tools are used for managing, evaluating, investigating, and analyzing the Infrastructure and services of the cloud computing environment. Similarly, cloud administrators or auditors monitor virtual resources, physical resources, applications, hosted data, and many heterogeneous cloud resources on the cloud. Cloud monitoring is considered a dominant part of the view of both service providers and consumers. It manages and controls the hardware and software setups on one side. In contrast, it offers statistics and key performance indicators (KPIs) for cloud platforms and applications to measure service effectiveness Alzakholi O, Haji L, Shukur H, et al (2020). Cloud monitoring tools follow the general

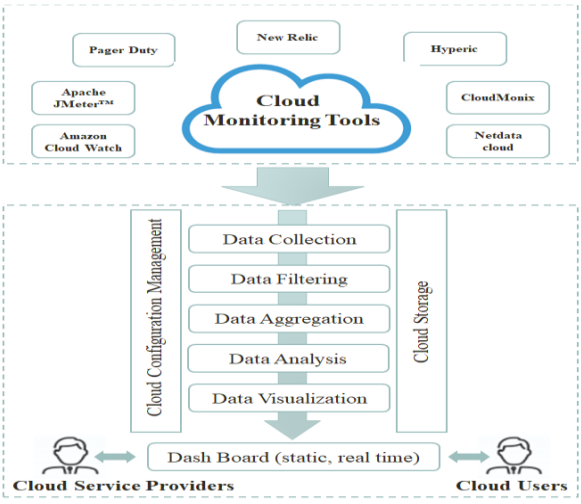

Fig.2: Cloud Monitoring Tools with its basic Architecture

architecture to analyze the resource (Birje & Bulla, 2019). Figure 2 illustrates the standard cloud monitoring tools with their monitoring steps. The cloud monitoring basic architecture involves data

gathering for analysis; data cleaning to remove redundancies; data aggregation to express data in the form of summary; data analysis used to review, transform, and model data; alerts/notifications are generated based on the data analysis; and finally, reports and visualization of data to make decisions. Additionally, cloud configuration settings are involved in each step of cloud monitoring tools and databases that store records. The dashboard shows all cloud resource monitoring information to customers and cloud providers.

Table 1 provides a summary of top cloud monitoring tools. The overview of cloud monitoring tools comprises distinct features such as cloud types, parent company, license, notifications, operating system, and properties to perform operations smoothly regarding SLA management, resource availability, privacy, and fault management.

Table 1: Cloud Monitoring Tools: Tabulated Summary

Tools	**Cloud Type**	**License**	**Operating System**	**Devices**	**Dashboard**	**Supported Languages**
Microsoft Cloud Monitoring (OMS) Keiko., H. (2021).	Hybrid Cloud	Commercial	Linux, Windows	Web-Based, Android, iOS	Real-time	. NET, Java, and Node. js
Amazon CloudWatch (AWS, 2022)	Private cloud	Commercial	Linux, Windows	Web-based	Real-time	PowerShell Windows, Perl for Linux
Jmeter (Apache JMeter™, 2021)	Public and Private	Open Source	Linux, Windows, Mac OSX	Desktop App.	Real-Time	Java, NodeJS, PHP, ASP.NET
Netdata.cloud (netdata, 2021)	Public and Private	Open Source	Linux, FreeBSD, and macOS	Web-based	Real-time	C, python, node.js, and bash
CloudMonix (Netro, 2021)	Hybrid cloud	Commercial	Linux, Windows	Web-based	Real-time	N/A
Hyperic(VMware, 2020)	Private Cloud	Open Source	Windows, Mac, Linux, and Unix	Web-Based	Real-time	N/A
AppDynamics(AppDynamics, 2021)	Public Cloud	Commercial	N/A	Web-based	Static	java, PHP, .NET and Node.js
New Relic(Relic, 2020)	Private Cloud	Commercial	Windows, Android, iOS	Web-based, Android, IOS	Real-time	C SDK, Go, , java,.NET , Node.js, PHP , Python, Ruby
Bitnami Stacksmith (Stacksmith, 2021)	Public and Private cloud	Commercial	Windows	Web-Based	Real-time	Python, Java, PHP, Go, Ruby, and Node
Unified Infrastructure Management (Broadcom, 2021)	Private cloud	Commercial	Windows, iOS, Android	Web-based, Android, iOS	Real-time	C SDK, JAVA SDK, and Perl SDK

3.1.2. QoS in Cloud Computing

Quality of services (QoS) is essential to the cloud environment to make cloud services more adaptable and acceptable for cloud customers. Cloud computing resources are disseminated internationally and provide promising features depending on the users' demand She Q, Wei X, Nie G, Chen D (2019).

Figure 3 provides the taxonomy of QoS in could computing with its parameters, techniques, and standards.

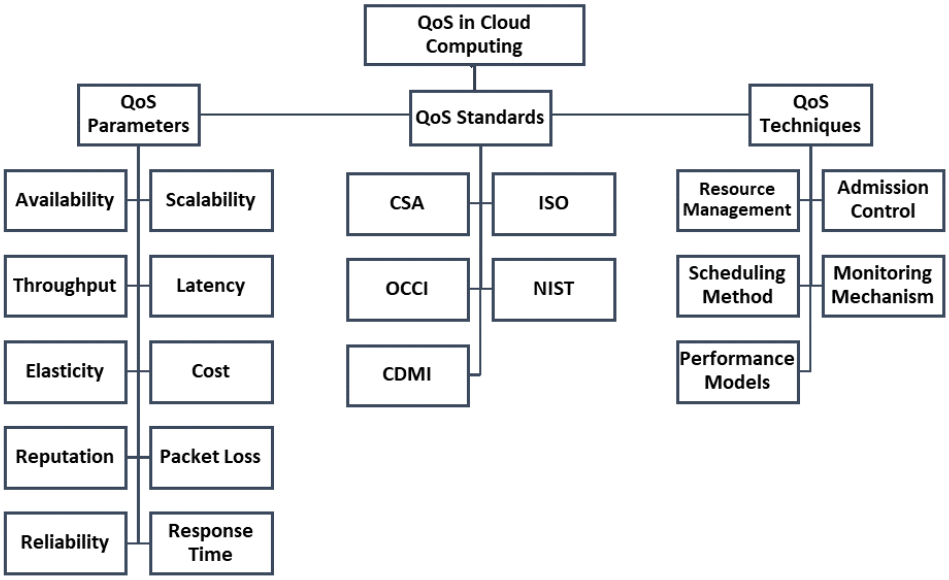

Fig.3: Taxonomy of Quality of Services (QoS) in cloud computing

3.1.3. QoS Parameters

When talking about the promising services of cloud computing, QoS parameters are involved. Different users require different QoS parameters according to their preferences. The QoS parameters could be availability, cost, throughput, latency, reliability, response time, elasticity, reputation, packet loss, and so on (Jelassi, Ghazel, & Saïdane, 2017), (Singh, Aggarwal, & Mishra, 2017), (Chitgar, Jazayeriy, & Rabiei, 2019) . The availability parameter measures the probability of that resources being available in total time. The parameter throughput denotes the maximum number of requests processed in a given time.

Furthermore, latency is the millisecond time for data processing packets to be transferred, stored, or retrieved by the computer machine. The parameter response time denotes the time the processor takes to deal with a process and submit the reaction. In addition, QoS should also provide scalability without upsetting the system's overall performance as the number and quantity of resources increase as required. In elasticity, the ability to remove or add resources dynamically as the workload changes, there should be a way to allow resources to be adapted automatically to reduce the infrastructure cost.

3.1.4. QoS Techniques

Many QoS modelling techniques are currently available, as seen in Figure 3. Researchers (Zanbouri & Jafari Navimipour, 2020) proposed the scheduling model to investigate user requirements and customize the workflow schedule. A fuzzy clustering method is used to classify the workflows. This scheduling model monitors QoS parameters such as bandwidth, response time, availability, reliability, and cost. Another research (Hassan, El-Desouky, Ibrahim, El-Kenawy, & Arnous, 2020) presented a trust management model constrained on QoS attributes and calculated the trust value. However, the authors did not clearly describe attribute combination and prioritization. In the work of (Farid, Latip, Hussin, & Abdul Hamid, 2020), authors presented an admission control framework based on different methods as horizontal elasticity and SLAs requirements. Furthermore, (Naseri & Jafari Navimipour,

2019) proposed a cloud monitoring framework for the public cloud to provide optimized resources and promise better performance.

3.1.5. QoS Standards

Many companies have adopted various standards with the theme of cloud computing. Figure 2.3 shows the major QoS standards commonly used in cloud computing. The QoS standards such as National Institute of Standards and Technology (NIST) (NIST, 2020), Open Cloud Computing Interface (OCCI) (OCCI-WG, 2016), Cloud Data Management Interface (CDMI)(, 2022), and Cloud Security Alliance (CSA) (Alliance, 2022) help build bridges to achieve interoperability and portability in cloud infrastructure. NIST launched the cloud computing program in November 2010 to save costs and implement safe and source enterprise practices (NIST, 2020). Furthermore, an open cloud computing interface (OCCI) is an Application Programming Interface (API) that acts as a front-end service provider to the IaaS management framework.

Moreover, the CDMI denotes the operational interface applications use to create, delete, retrieve, and update data elements in the cloud environment. OpenStack is an open-source IaaS cloud management platform that are a collection of software modules and tools that provides a framework to create and manage both public cloud and private cloud (Lima, Rocha, & Roque, 2019). Similarly, the CSA (Alliance, 2022) and the international standards organization (ISO) work together and recognize the cloud security and privacy standards.

3.1.6. Cloud QoS Violation Detection

3.1.7. QoS Violation Detection

Service level agreement (SLA) guarantees SLA definition; basic structure with quality of services (QoS); SLA negotiation and monitoring; detection of SLA violations and enforcements. With the acceptance of SLA terms, the QoS plays a significant role and provides promising services. Similarly, QoS is documented as the SLA that stipulates the commitment between consumers and service providers and monetary penalties in case of SLA violations. Hence, cloud service providers need to detect and predict possible QoS violations. However, it is difficult to deal with service violations due to several QoS parameters in a cloud-based environment. In literature, many techniques are used to detect and predict QoS violations. These techniques are discussed in the following subsections.

3.1.8. QoS Violations Detection Techniques

This section summarizes the QoS violation detection techniques with their substantial attributes, i.e., methodology, research model, techniques used, parameters, findings, and future work, as shown in Table 2. For example, research (Agarwal, 2020), (Hani & Paputungan, 2017), (Anitha & Vidyaraj, 2019) had considered the various QoS parameters such as availability, response time, throughput, storage memory,and CPU utilization.

The work of (Hani & Paputungan, 2017) implemented the Support Vector Regression model to identify and detect violations from the cloud environment by considering QoS parameters such as availability, response time, and throughput. The QoS violations are categorized into various levels based on severity; for example, when there is no violation, then no action is taken. However, when violations occur at a low and medium level, the model shall determine them as no re-negotiation or renegotiation, respectively. When the severity of the violation is detected to be high, the model declares it as a re-negotiate and imposes penalty charges. Consequently, QoS metrics such as availability are categorized into daily and night availability due to the varying loads at different times.

Similarly, another QoS parameter, response time, is split into day and night response times. This manifold learning model changed the way how data is being treated. The model could convert high-

dimensional data into low-dimensional datasets easily. However, further research is required to improve detection and prediction accuracy by inserting more QoS parameters and deploying other techniques (Biswas, Banerjee, Biswas, & Ghosh, 2021).

Another research (Agarwal, 2020) implemented the Naive Bayes (NB) and Random Forest (RF) machine learning techniques for data analysis, QoS violation detection, and design optimization. The unified framework successfully detects and prevents QoS violations and considers response time, CPU, and memory utilization parameters. Despite that, the system's performance can be enhanced by inserting more training datasets and other machine learning techniques. In (Hemmat & Hafid, 2016), a method based on the machine learning classifiers Naive Bayes (NB) along with Random Forest (RF) models are proposed to detect SLA violations on the realworld dataset. Twenty-nine days of Googles Cloud compute trace are used for the experiment, and the dataset is obfuscated due to security reasons. Consequently, the proposed model could achieve up to 99.88% accuracy with the Random Forest machine learning classifier.

3.2. Cloud QoS Violation Remediation

This section introduces the remedial action to be implemented while QoS violation in cloud infrastructure. Generally, the scalability of resources is used for preventive and remedial action, while fault tolerance is used as a defensive measure.

3.2.1. Resource Scalability

The objective of scalability is to maintain the performance of running resources on the cloud to steer the cloud resources away from QoS violation. Scalability deals with the growing workload by assigning more resources or services to the system. Figure 4 demonstrates two resource scalability methods in a cloud computing environment.

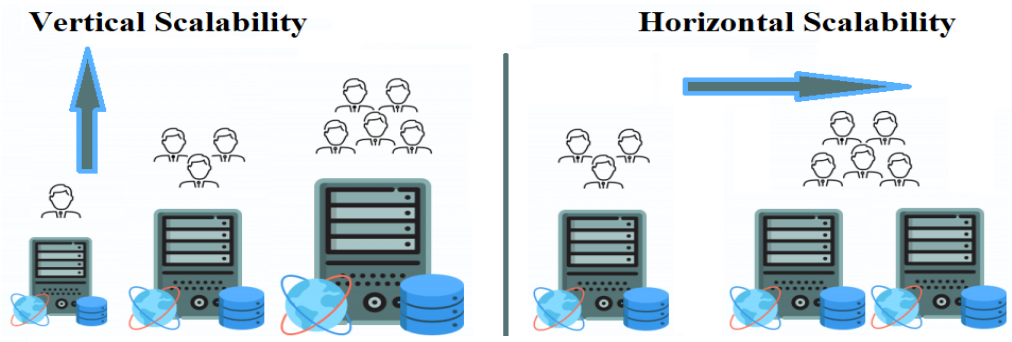

Fig.4: Resource scalabilities techniques in cloud computing

There are two categories of scalability approaches: vertical scaling and horizontal scaling. In vertical scaling, the capacity of the resources, i.e., RAM or storage, CPU, and many other resources of the present system, is increased or decreased to meet the required performance state. Under horizontal scaling, more resources are added to the resources pool to share the existing workload between the resources (Wong, Chan, & Chua, 2019). Furthermore, multiple resource allocation methods are applied in the literature to deal with the inefficiency in resource assigning and increasing the power consumption ratio. For example, Research by (Kardani-Moghaddam, Buyya, & Ramamohanarao, 2020) proposed Hybrid Anomaly-aware Deep Reinforcement Learning-based Resource Scaling (ADRL) for dynamic cloud resource scaling. ADRL uses anomaly detection to trigger actions to stabilize decisionmakers. Scaling requires global and local decisionmakers. research by (Rai et al., 2021) suggested a vertical

scaling framework for computing infrastructure. In this study, Hypervisor VirtualBox is utilized to execute the Vertical Scaling using Command Line Interface (CLI). A Python script is developed to manage vertical VM scaling. During the process, statistics regarding user activity are logged to the database with any changes made to the VMs. A web interface is made available to the user and the administrator so that they may monitor the changes made to the VM environment across several user sessions. Scaling of various VM factors such as RAM and VRAM size, utilization of several CPUs, and availability is ensured. The authors presented a cost-effective and efficient strategy for implementing vertical scaling of virtual machines within cloud infrastructure.

Bruno et al. (2018) introduced a novel JVM heap-sizing approach that allows JVM to dynamically scale up its memory consumption based on the application's requirements. In the study (Shahin, 2017), an auto-scaling algorithm with a dynamic threshold to predict requisite resources using the "Long Short-Term Memory Recurrent Neural Network" (LSTM-RNN) is proposed. The developed algorithm was implemented on the Cloudsim simulator and a deep learning library such as Deeplearning4j.

Research by (Shelar, Sane, & Kharat, 2016) proposed a hybrid scaling method to increase resource scalabilities such as memory, storage, CPU, and network bandwidth from cloud providers. Persico et al. (2017) offered a new Fuzzy-PID architecture to inevitably perform cloud resources scalability at the virtual machine granularity by considering the heterogeneous parameters. The suggested architecture aims to maintain resource scaling and guarantee SLAs. Parameters used include the computational and the network ability to deploy the presented architecture. The feedback control is handled through proportional integrative derivative (PID) and fuzzy logic used to gain control to predict the management techniques by cloud providers. Research by Hui et al. (2018) proposed a novel "Elastic Network Service Chain" (ENSC) model that employed a hybrid scaling technique to accomplish scalability and NFV efficiency. The authors delivered Rubik's heuristic algorithm and conveyed the resource allocation problem with integer linear programming (ILP) in cloud datacenters. The experiment result reveals that ENSC attains the highest acceptance rate and resource consumption compared to horizontal and vertical scaling techniques, FreeFlow ElasticNFV (Yu, Yang, & Fung, 2018) respectively.

In Table 2, studies related t resource scalability techniques are present.

Table 2: Resource scalability Techniques: Tabulated Summary

References	**Method/Approach**	**Type of Scalability**	**Description**
(Kardani-Moghaddam et al., 2020)	Hybrid Anomalyaware Deep Reinforcement Learning-based Resource Scaling (ADRL) for dynamic cloud resource scaling	Hybrid Scaling	ADRL uses anomaly detection to trigger actions to stabilize decision-makers. Scaling requires global and local decision-makers. Each VM collects resource utilization metrics periodically to monitor resource performance. Local Data Analyser (DA) and Reinforcement Learning (RL) agents process collected data.
(Yu, Yang, Fung, et al., 2018)	The ENSC approach provides hybrid scaling to attain scalability and NFV efficiency.	Hybrid Scaling	A simulated cloud data center consists of the 3-level tree topology, 1600 servers, one core switch, 80 ToR, and four aggregation switches for the experimental setup. Similarly, homogenous PM with CPU (12 vCPUs), a memory of 32 GB, and a bandwidth capacity of 1 Gbps. The l results show ENSC achieved the highest acceptance ratios in hybrid scaling.
(Bruno et al., 2018)	The novel JVM heap sizing strategy aims to dynamically	Vertical Scaling	The evaluation techniques, such as

	scale up the memory consumption according to the processes needed.		OpenJDK, HotSpot, and JVM 9 are deployed. Furthermore, the DaCapo benchmark suite and the Tomcat web server are compared with the JVM heap sizing strategy.
(Persico et al., 2017)	An approach for horizontal scalability is presented to scale cloud resources to meet irregular and time-fluctuating operating conditions on the public cloud.	Horizontal Scaling	From the experimental point of view, a test bed is composed of these three elements: 1. cloud services implemented on the public cloud 2. master node to host all blocks of the architecture 3. emulation node to issue the requests to cloud applications
(Shahin, 2017)	An auto-scaling algorithm (LSTM-RNN) based on the dynamic threshold to forecast the required number of resources is proposed.	Auto-scale Virtual Scaling	For the experimental purpose, the CloudSim simulator and real traces are used.
(Shelar et al., 2016)	the main goal is to scale resources dynamically by considering responsiveness, revenue, and availability of resources.	Hybrid Scaling	For the experimental purpose of CPU pool utility in the XL tool stack, Libvirt package and XEN hypervisor are used. As a result, the delivered hybrid scaling technique successfully worked and fulfilled the demand for additional resources and avoided resource migration.

3.2.2. Resource Allocation Techniques

Resource allocation is a process that is used to dispense manageable cloud resources to the resource-seeking cloud applications accessible through the Internet in an organized way. There are two main parties in a cloud environment: cloud service providers with a pool of resources in their datacentre and cloud service users that demand the resources from the service providers. This way, cloud providers rent out their resources on a pay-per-use policy and generate maximum revenue from cloud users, as depicted in Figure 5. The process of resource allocation consists of three steps. In the initial step, clients request cloud resources from cloud vendors such as Oracle, AWS (Amazon, 2022), and Azure. The second step ensures the availability of cloud recourses to customers, and then finally, in the third step, the client utilizes the resources. Users want cost-effective and more efficient resources to complete the specific task in an optimized time. In cloud computing, resource management is entirely based on resource allocation.

Authors (Lai, et al., 2020) proposed a heuristic approach to solve the QoE-aware "edge user allocation" (EUA) problem with "Integer Linear Programming" (ILP) and identified the optimal

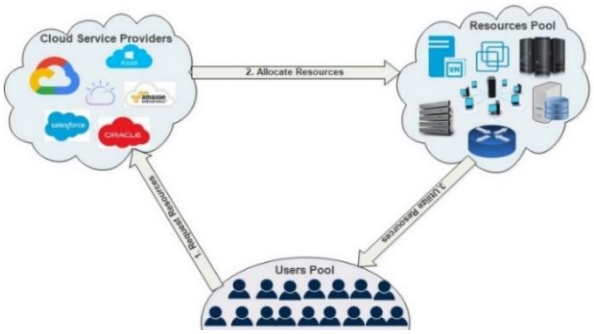

Fig.5: The basic process of resource allocation

solution. A real-world dataset is utilized for the experiment The authors utilized the Antlion optimization approach. Further it is divided into three modes: 1) resource allocation, 2) Client requests and shared resources are scheduled., and 3) finally, the design of the presented algorithm. The methodology's objective is to reduce resource utilization and ensure the optimized allocation of resources. RAAJS algorithm is evaluated on the cloud test-bed and SimGrid toolkit used for simulation.

Another author (Bakalla, Al-Jami, Kurdi, & Alsalamah, 2017) introduced the genetic algorithm for resource allocation (GARA) to detect SLA violations and maintain energy utilization. Furthermore, the CloudSim toolkit was used to evaluate the efficiency of the GARA policy. A simulation environment is considered that maintains eight hundred heterogeneous physical machines (PM), and two servers are also configured. GARA is compared with existing approaches such as NonPowerAware and the "Dynamic Voltage Frequency Scaling" (DVFS) algorithm for performance analysis. These two algorithms compared the energy consumption and did not consider SLA violations.

3.2.3. Resource Allocation using Artificial Intelligence

Wu et al. (Wu, Wang, Wei, & Shi, 2020) developed a hybrid adaptive prediction model to estimate VM load using CPU and memory metrics. It modified resource settings based on load. Authors configure VMs to adapt to user needs to fix dynamic VM resource allocation. Flow conservation and link usage result from the analogous optimization issue. The exact answer and a rapid suboptimal solution are proposed. Accuracy and optimization time are compared. The proposed solution reduced packet loss by 20%, increased throughput by 30%, and used CPU and memory by 70%. It can't estimate CPU demand and has to be tested further. The method (Praveenchandar & Tamilarasi, 2020) involves three phases. First, "Preference-Based Task Scheduling" (PBTS) dynamically allocates resources based on client entries. A dynamic resource table (DRA) keeps track of available resources. It updates customers' and providers' resource allocation requests and responses. Customers can verify resource availability.

A study (Khan, H. M., Chua, F. F., & Yap, T. T. V., 2022) presents the resource scalability method by considering the QoS parameter values, response time, and throughput. Based on QoS parameter bounds, cloud service is classified for violations. If QoS violations occur, then quantified resources are allocated to the VM to rectify them. It has been studied that there is a nonlinear relationship between workload and cloud resources. ReSQoV emphasizes cloud QoS violations caused by under- or over-provisioning by resource use estimate. The suggested method is based on the Universal Scalability Law (USL), which helps forecast a specific load's service capacity. The model coefficients represent the quantitative values of contention and coherence, while the p-values indicate the model's fit. J. Zhang et al. (2020) developed a unique integer programming model to handle the cloud's time-varying multidimensional resource allocation and pricing systems.

Naha et al. (Naha, Garg, Chan, & Battula, 2020) created the deadline-based "Resource Ranking and Provisioning" (ReRaP) algorithm to deal with fluctuating user demand. The resource allocation and scheduling process are based on searching for available resources to meet deadline requests under three conditions: high computational resources, low network bandwidth, and cloud response time. The experiment is conducted with a CloudSim simulator. Chen et al. (X. Chen, Wang, Ma, Zheng, & Guo, 2020) suggested a QoS-based self-adaptive approach to assigning cloud resources. First, the model is dynamically tweaked utilizing self-tuning control. Second, a feedback loop is created to allot resources based on unnecessary overhead. The authors compared the model's performance to RUBiS. The findings showed that the suggested approach improved QoS predictions and resource allocation. The elastic cloud can handle fluctuating workloads and consumption. (Geekbench, 2022) Researchers provided cloud application resource provisioning via workload clustering. BBO and K-means clustering was used to divide cloud workloads by QoS. Bayesian learning was utilized to identify QoS-compliant

resource provisioning activities. The suggested resource provisioning strategy reduces time, SLA violation percentage, cost, and energy utilization.

Autonomous and smart cloud capacity management, resource allocation, and provisioning solutions are sought. The study (Ghobaei-Arani, 2021) offered a resource strategy based on smart agents and cloud user services. The technique emphasizes low-cost maximum resource use and QoS guarantee with little VM commitment. Optimal fit provides the best cost-to-VM-placement ratio for service providers and customers. The suggested approach optimizes VM resource allocation, flexibility, scalability, and dependability in datacenters for performance and energy consumption.

4. Analysis and Discussion of the State-of-the-Art Works

After the review, the evaluation and comparison of works related to QoS metrics, monitoring, tools, and violation detection and remediation in cloud computing are presented in terms of existing methods, their capabilities and limitations, and the proposed solutions.

4.1. Cloud QoS Monitoring

Various databases collect state-of-the-art literature in identifying QoS metrics and monitoring in cloud services. It is found that QoS is a non-functional requirement for cloud services and is agreed upon in the SLA between the cloud service providers and cloud service consumers. Cloud service performance is evaluated using service availability, response time, and throughput typically agreed upon in SLA to ensure the QoS. The performance monitoring tools are part of the service provider's infrastructure that provides resource usage but does not focus on the cloud user QoS. A list of monitoring tools is identified for public and private cloud that monitors infrastructure resources and a few on the cloud service performance. We have determined that third-party performance monitoring tools can help in QoS and performance evaluation of cloud services. Table 3 presents the QoS monitoring tools, applications on cloud service types, license types, and overall analysis of the tool's useableness.

Table 3: Summarizes and compares the QoS monitoring tools.

Source	**Cloud Type**	**License Type**	**Analysis**
Azure Monitor (Azure, 2022)	Hybrid Cloud	Commercial OMS Discon. (2019)	•Cloud Service providers' monitoring tools are service utility and environment specific. •Mainly third-party monitoring tools are commercial, and their monitoring ops, parameters, and reports are specific. •Generally, monitoring tools focus on infrastructure monitoring and lack the service performance monitoring
Amazon CloudWatch (AWS, 2022)	Private cloud	Commercial	
Jmeter (Apache JMeter™, 2021)	Public and Private	Open Source	
New Relic (Relic, 2020)	Hybrid cloud	Commercial	
Nagios (Nagios, 2022)	Hybrid Cloud	Commercial	
CloudMonix (Netro, 2021)	Hybrid cloud	Commercial	
vRealize Hyperic (VMware, 2020)	Private Cloud	Open Source (Discontinue)	

The analysis of QoS monitoring tools suggests that monitoring tools are available from public cloud providers and their monitoring services are limited to their Platform (i.e., (AWS(b), 2021)). Few monitoring tools are available for public and have open source licence (i.e., (JMeter™, 2017))for private cloud.

For the QoS cloud service monitoring, the adoption/selection of open source, multi-threaded monitoring application for desired QoS parameters and desired data gathering for Cloud service workload generation is more feasible and cost effective.

4.2. Cloud QoS Violation Detection

This section presents the analysis of collected studies related to the problem of QoS violation detection. Effective cloud resource monitoring mechanisms are needed to ensure that the software and hardware deployed by the cloud service providers are running at a satisfactory quality level as agreed in the SLA. This mechanism should be able to gather QoS parameter values and help to detect QoS violations. The aim is to find contributions from existing literature on how the QoS violation detection problem is being solved, and here the studies are analyzed accordingly. Table 4 presents the violation detection techniques and analysis.

Table 4: Violation Detection Techniques and QoS Parameter

Source	**QoS parameters**	**Violation Detection Techniques**	**Analysis**
Chauhan, N., & Agrawal, R., 2022	CPU utilization, response time, and memory utilization.	Naive Bayes	• There is a lack of an adaptive solution for the variant nature of cloud services. • There is a need to emphasize QoS parameters rather than infrastructure parameters • The performance of models prediction results and accuracy is needed to be improved, i.e., ANFIS RMSE 1.3% (Ghobaei., et al., 2019), SVR Accuracy 80% (Hani, A. F. M., et., at., 2017)
Agarwal, 2020	Response time, Memory, and CPU utilization	Naive Bayes (NB) and Random Forest (RF)	
Etemadi, et al., 2020	CPU utilization, cost	Bayesian learning	
Ghobaei et al., 2019	Response Time, Cost	ANFIS	
Aslanpour et al., 2018	Response Time, SLA Violations, Cost	Radial basis function neural net (RBFNN)	
Hani, A. F., et at., 2017	Availability, response time, and throughput	Support Vector Regression model	

This literature review and comparison of QoS violation detection techniques shows in Table 2.4 that there are multiple QoS parameters used for violations detection. Machine learning techniques, that include Nauive Bayes, Random Forest, ANFIS, Support Vector Regression model etc, are considered by the researcher for violation detection and their performance and adaptability varies. It also found space for improving QoS parameter selection using gap identification to provide satisfied QoS. The performance of Violation detection accuracy can be improved.

4.3. Cloud QoS Violation Remediation

Resource allocation and scalability have been considered potential remedial actions for QoS violations. Therefore, managing and allocating resources adequately to avoid QoS violations is among the most

challenging tasks in the current cloud computing research. Consequently, many researchers have proposed techniques that address this QoS violation problem in the cloud environment. Scalability is essential as it is related to increased workload and allocation of cloud resources to the system. Scalability could be determined by the available resources and how the data flow of applications or services is controlled. Improper control of such data flow would lead to under-provisioning, resulting in high response times or low throughput or the event of over-provisioning, high costs, with low utilization of resources. Table 5 presents the resource allocation methods in the literature and the analysis extracted.

The studies show that researchers used various techniques to solve the resource allocation: energy-efficient, multidimensional, self-adaptive, cost-aware, dynamic, deadline-based, load balancing, and Artificial intelligence heuristics, algorithms, and models. The findings and limitations describe that each method focuses on some specific issue, i.e., user load, energy consumption, or throughput. The literature analysis suggests that QoS violation remediation should emphasize scalability overheads and that the nonlinear relationship between workload and resources can improve performance. The literature review findings also describe the need to improve the methods concerning QoS and performance.

Table 5: Resource Allocation Methods

Source	**Resource Allocation Methods**	**Description**	**Analysis**
Khan, H. M. et al, (2022)	Response time and throughput	ANFIS, with 16 rules	Few studies focused on the non-linearity consideration among workload and cloud resources. Khan, H. M. et al, (2022) This problem is observed in some studies results, but no discussion and consideration performed (Wu et al., 2020) • No specific emphasis on scalability overhead • Efficiency and performance evaluation of the models can be improved, i.e., SVM 88.9%, CART 76.8% (Chen et al., 2020)
Wu et al., 2020	Energy-efficient resource allocation	An optimized solution for VM workload	
Zhang et al., 2020	Multidimensional resource allocation	Optimized resource allocation and benefits for resource providers	
Naha et al., 2020	Deadline-based dynamic resource allocation	To deal with the dynamic behaviour of users in terms of deadline, response time, and budget with accuracy.	
Chen et al., 2020	Iterative Self-adaptive resource allocation	Balanced cost of resources with QoS using particle swarm optimization (PSO) optimization	
Gill, S. S., et al., 2017	QoS-based autonomic resource management approach	The approach considers Configuring, Healing, Optimizing, and Protecting Policy for Efficient autonomic resource management	

5. Validity Threats

Several threats might jeopardize the validity of literature review research. Prominent guidelines and directions were taken into consideration in this work to mitigate validity risks, as follows:

Research questions addressed: This study may not incorporate all current research aspects of cloud QoS monitoring, violation, and remediation. To address this issue, all researchers collaborated to identify the most recent research topics in the field.

Review of relevant papers: The method of gathering all relevant research on cloud QoS violations and remediation cannot be guaranteed. In this research, various literature databases are used, and all writers used the approach based on distinct phrases and synonyms to determine the associated questions.

Paper inclusion /exclusion criteria: Individual prejudice and interpretation may impact how the criteria are implemented. To address the validity issue, all authors' agreements were considered when omitting or adding an article.

The study's reproducibility: Another risk is that other researchers might replicate the findings of this study. As a result, the research methodology includes the well-explained procedures and activities used in this work.

6. Conclusion and Future Directions

Trust management plays a vital role in the cloud environment, through which cloud acceptability increases and providers generate customer value and revenue. In this review paper, we have explored the field of Quality of Service (QoS) monitoring, violation detection, and remediation for cloud computing. The rapid growth and adoption of cloud computing have necessitated the development of effective techniques and frameworks to ensure QoS in cloud-based systems. We began to examine studies related to the concepts of QoS monitoring, violations, and remediation using resource allocation and scalability in cloud computing. We discussed a wide range of QoS metrics that are essential for evaluating and maintaining the performance of cloud services. The review presented an in-depth analysis of existing approaches for QoS monitoring in the cloud, ranging from infrastructure-level monitoring to application-level monitoring. Various monitoring tools and technologies were examined, showcasing the diverse methods employed to collect and analyse QoS data. Furthermore, we explored techniques for detecting QoS violations in the cloud environment. Machine learning algorithms, and hybrid methods were discussed, shedding light on the potential solutions for accurately identifying QoS breaches. In addition to violation detection, we investigated remediation strategies for addressing QoS violations in the cloud. Proactive and reactive approaches were presented, including QoS-aware, energy-efficient, dynamic, multidimensional deadline-based, load balancing, cost-aware, workload, scalability, and adaptive resource allocation. We discussed the challenges associated with these techniques effectively and highlighted the need for further research in this area.

Overall, this review paper serves as a comprehensive resource for researchers involved in QoS assurance for cloud-based systems. It consolidates the current knowledge and advancements in QoS monitoring, violation detection, and remediation, providing insights into the strengths and limitations of existing approaches. As cloud computing continues to evolve, it is imperative to develop standardized frameworks and benchmarks for evaluating and comparing QoS monitoring and remediation techniques. Additionally, future research should focus on addressing the emerging challenges in dynamic and multi-tenant cloud environments, such as real-time QoS monitoring and adaptive remediation strategies. By enhancing QoS assurance in the cloud, the reliability and performance of cloud services can be improved, ultimately leading to increased user satisfaction and trust. Continued advancements in QoS monitoring, violation detection, and remediation will play a crucial role in shaping the future of cloud computing and enabling the delivery of high-quality services to users worldwide. While the review paper provides a comprehensive analysis of the current state-of-the-art techniques and frameworks, it

is important to acknowledge some limitation and validity threats to enhance the credibility and reliability that include research questions addressed, review of relevant studies and search strategy.

Acknowledgement

This research was funded by Multimedia University, Malaysia.

References

Agarwal, S. (2020). An approach of sla violation prediction and qos optimization using regression machine learning techniques (Unpublished doctoral dissertation). University of Windsor (Canada).

Alliance, C. S. (2022). *Cloud security alliance (csa), awareness of best practices to help ensure a secure cloud computing environment.* Retrieved 2022-02-25, from https:// cloudsecurityalliance.org/

Amazon. (2022). *Amazon web services, inc. amazon ec2 instance types—amazon web services.* Retrieved 2022-02-20, from https://aws.amazon.com/ec2/instance-types/

AppDynamics. (2021). *Manage new demands in a multi-cloud world.* Retrieved 2022-03-26, from https://www.appdynamics.com/

Aslanpour, M. S., Dashti, S. E., Ghobaei-Arani, M., & Rahmanian, A. A. (2018). Resource provisioning for cloud applications: a 3-d, provident and flexible approach. *The Journal of Supercomputing*, *74*(12), 6470–6501.

AWS, (2022). *Amazon cloud watch.* Retrieved 2022-03-26, from https://aws.amazon.com/ cloudwatch/

Bakalla, M., Al-Jami, H., Kurdi, H., & Alsalamah, S. (2017). A qos-aware and energy-efficient genetic resource allocation algorithm for cloud data centers. In *2017 9th international congress on ultra modern telecommunications and control systems and workshops (icumt)* (pp. 244–249).

Birje, M. N., & Bulla, C. (2020). Commercial and open source cloud monitoring tools: a review. *Advances in Decision Sciences, Image Processing, Security and Computer Vision*, 480–490.

Bokhari, M. U., Shallal, Q. M., & Tamandani, Y. K. (2016). Cloud computing service models: A comparative study. In *2016 3rd international conference on computing for sustainable global development (indiacom)* (pp. 890–895).

Bruno, R., Ferreira, P., Synytsky, R., Fydorenchyk, T., Rao, J., Huang, H., & Wu, S. (2018). Dynamic vertical memory scalability for openjdk cloud applications. In *Proceedings of the 2018 acm sigplan international symposium on memory management* (pp. 59–70).

CDMI (2022). Cloud data management interface (cdmi). Retrieved 2022-8-25, from https://www.iso.org/standard/83451.html

Challagidad, P. S., & Birje, M. N. (2017). Trust management in cloud computing. In *2017 international conference on smart technologies for smart nation (smarttechcon)* (pp. 295–298).

Chauhan, N., & Agrawal, R. (2022). Probabilistic optimized kernel naive bayesian cloud resource allocation system. *Wireless Personal Communications*, 1–20.

Chen, X., Wang, H., Ma, Y., Zheng, X., & Guo, L. (2020). Self-adaptive resource allocation for cloud-based software services based on iterative qos prediction model. *Future Generation Computer Systems*, *105*, 287–296.

Chitgar, N., Jazayeriy, H., & Rabiei, M. (2019). Improving cloud computing performance using task scheduling method based on vms grouping. In 2019 27th iranian conference on electrical engineering (icee) (pp. 2095–2099).

Etemadi, M., Ghobaei-Arani, M., & Shahidinejad, A. (2020). Resource provisioning for iot services in the fog computing environment: An autonomic approach. *Computer Communications*, *161*, 109–131.

Farid, M., Latip, R., Hussin, M., & Abdul Hamid, N. A.W. (2020). A survey on qos requirements based on particle swarm optimization scheduling techniques for workflow scheduling in cloud computing. Symmetry, 12(4), 551.

Geekbench. (2022). *Geekbench processor benchmarks.* Retrieved 2022-02-20, from https:// browser.geekbench.com/v5/cpu

Ghobaei-Arani, M. (2021). A workload clustering based resource provisioning mechanism using biogeography based optimization technique in the cloud based systems. *Soft Computing*, *25*(5), 3813–3830.

Gill, S. S., Chana, I., Singh, M., & Buyya, R. (2018). Chopper: an intelligent qos-aware autonomic resource management approach for cloud computing. *Cluster Computing*, *21*(2), 1203– 1241.

Hani, A. F. M., & Paputungan, I. V. (2017). Manifold learning in sla violation detection and prediction for cloud-based system. In *Proceedings of the second international conference on Internet of things, data and cloud computing* (pp. 1–5).

Hassan, H., El-Desouky, A. I., Ibrahim, A., El-Kenawy, E.-S. M., & Arnous, R. (2020). Enhanced qos-based model for trust assessment in cloud computing environment. IEEE Access, 8, 43752–43763.

Hayyolalam, V., Pourghebleh, B., & Pourhaji Kazem, A. A. (2020). Trust management of services (tmos): investigating the current mechanisms. Transactions on Emerging Telecommunications Technologies, 31(10), e4063.

Hemmat, R. A., & Hafid, A. (2016). Sla violation prediction in cloud computing: A machine learning perspective. *arXiv preprint arXiv:1611.10338*.

Jelassi, M., Ghazel, C., & Saïdane, L. A. (2017). A survey on quality of service in cloud computing. In *2017 3rd international conference on frontiers of signal processing (icfsp)* (pp. 63–67).

JMeter™, A. (2017). *Apache jmeter load test functional behavior and measure performance of static and dynamic resources.* Retrieved 2022-4-20, from https://jmeter.apache.org/index .html

Kan, Y. (2021). A cloud computing resource optimal allocation scheme based on data correlation analysis. In *The 4th international conference on electronics, communications and control engineering* (pp. 26–31).

Khan, H. M., Chua, F. F., & Yap, T. T. V. (2022). ReSQoV: A Scalable Resource Allocation Model for QoS-Satisfied Cloud Services. Future Internet, 14(5). https://doi.org/10.3390/fi14050131

Keiko., H. (2021). Hybrid cloud monitoring is easy with microsoft oms. Retrieved 2021-08-24, from https://cloudblogs.microsoft.com/opensource/2017/01/23/microsoft-operations-management-suite-hybrid-cloud-monitoring-easy/

Kohlbrenner, N. (2021). *Gdpr checklist: How to comply saas for eu data privacy.* Retrieved 2022-04-30, from https://www.cloudways.com/blog/saas-gdpr-checklist

Lai, P., He, Q., Cui, G., Xia, X., Abdelrazek, M., Chen, F., ... Yang, Y. (2020). Qoe-aware user allocation in edge computing systems with dynamic qos. *Future Generation Computer Systems*, *112*, 684–694.

LogicMonitor. (2021). *Your total it infrastructure. one monitoring platform.* Retrieved 2022-04-24, from https://www.logicmonitor.com/

Naha, R. K., Garg, S., Chan, A., & Battula, S. K. (2020). Deadline-based dynamic resource allocation and provisioning algorithms in fog-cloud environment. *Future Generation Computer Systems*, *104*, 131–141.

Naseri, A., & Jafari Navimipour, N. (2019). A new agent-based method for qos-aware cloud service composition using particle swarm optimization algorithm. Journal of Ambient Intelligence and Humanized Computing, 10, 1851–1864.

Netdata. (2021). *Monitor everything in real time – for free.* Retrieved 2022-04-20, from https:// www.netdata.cloud/

Netro. (2021). *Cloud and on-premise monitoring and automation.* Retrieved 2022-03-20, from https://cloudmonix.com/

NIST, Q. (2020). Nist, computer security resource center (csrc). information technology laboratory, glossary (sources: Nist sp 800-113, nist sp 800-125b, nist sp 800-187, nist sp 800-209, nist sp 800-77 rev. 1, nist sp 800-95). Retrieved 2021-08-24, from https://csrc.nist.gov/glossary/term/Quality_of_Service

OCCI-WG. (2016). Cocci 1.2 - open cloud computing interface – infrastructure. Retrieved 2022-3-25, from https://ogf.org/documents/GFD.224.pdf

Praveenchandar, J., & Tamilarasi, A. (2021). Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. *Journal of Ambient Intelligence and Humanized Computing*, *12*(3), 4147–4159.

Relic, N. (2021). *Uptime is everything.* Retrieved 2022-03-20, from https://newrelic.com/

Salesforce. (2021). *Salesforce.* Retrieved 2022-03-20, from https://www.salesforce.com/in/ ?ir=1

Shahin, A. A. (2017). Automatic cloud resource scaling algorithm based on long short-term memory recurrent neural network. *arXiv preprint arXiv:1701.03295*.

She Q, Wei X, Nie G, Chen D (2019) QoS-aware cloud service composition: A systematic mapping study from the perspective of computational intelligence. Expert Syst Appl 138:112804. https://doi.org/10.1016/j.eswa.2019.07.021

Shelar, M., Sane, S., & Kharat, V. (2016). Enhancing performance of applications in cloud using hybrid scaling technique. *International Journal of Computer Applications*, *143*(2), 0975–8887.

Singh, P., Dutta, M., & Aggarwal, N. (2017). A review of task scheduling based on meta-heuristics approach in cloud computing. *Knowledge and Information Systems*, *52*(1), 1–51.

vmware. (2020). *vrealize hyperic.* Retrieved 2022-01-25, from https://www.vmware.com/ products/vrealize-hyperic.html

Wong, T.-S., Chan, G.-Y., & Chua, F.-F. (2019). Adaptive preventive and remedial measures in resolving cloud quality of service violation. In *2019 international conference on information networking (icoin)* (pp. 473–479).

Wu, X., Wang, H., Wei, D., & Shi, M. (2020). Anfis with natural language processing and gray relational analysis based cloud computing framework for real time energy efficient resource allocation. *Computer communications*, *150*, 122–130.

Yakubu IZ, Musa ZA, Muhammed L, et al (2020) Service Level Agreement Violation Preventive Task Scheduling for Quality of Service Delivery in Cloud Computing Environment. Procedia Comput Sci 178:375–385. https://doi.org/10.1016/j.procs.2020.11.039

Yu, H., Yang, J., Fung, C., Boutaba, R., & Zhuang, Y. (2018). Ensc: multi-resource hybrid scaling for elastic network service chain in clouds. In *2018 ieee 24th international conference on parallel and distributed systems (icpads)* (pp. 34–41).

Zanbouri, K., & Jafari Navimipour, N. (2020). A cloud service composition method using a trustbased clustering algorithm and honeybee mating optimization algorithm. International Journal of Communication Systems, 33(5), e4259.

Zhang, J., Yang, X., Xie, N., Zhang, X., Vasilakos, A. V., & Li, W. (2020). An online auction mechanism for time-varying multidimensional resource allocation in clouds. *Future Generation Computer Systems*, *111*, 27–38.