AI-Driven *QoS* Optimization in *Multi-Cloud* Environments: Investigating the Use of AI Techniques to Optimize *QoS* Parameters Dynamically Across Multiple Cloud Providers

KAUSHIK SATHUPADI [1]

[1] *Staff Engineer, Google LLC, Sunnyvale, CA*

Published: 2022

Abstract

As businesses increasingly move towards multi-cloud environments for unique benefits of different cloud service providers (CSPs), ensuring optimal Quality of Service (QoS) becomes critical. QoS in multi-cloud environments involves balancing numerous parameters such as latency, throughput, availability, and resource allocation across multiple platforms. This paper explores the use of machine learning (ML) and deep learning (DL), for the dynamic optimization of QoS in multi-cloud environments. AI offers assistance to manage large-scale datasets, adapt to changing conditions, and learn from previous performance data to make intelligent decisions. The study focuses on how these AI techniques can minimize Service Level Agreement (SLA) violations, optimize resource usage, and enhance service reliability. The study investigates AI-driven approaches, such as reinforcement learning, neural networks, and predictive analytics to look into how automation in multi-cloud management can result in better resource efficiency, improved QoS, and reduced operational costs. This paper also discusses the challenges inherent in AI-driven multi-cloud management, such as data heterogeneity, system scalability, and security concerns. The application of AI to assist multi-cloud environments through real-time decision-making and predictive modeling is emphasized, showing how these technologies can transform the future of cloud computing infrastructure.

1. INTRODUCTION

As cloud computing has become an integral part of modern enterprise IT infrastructure, the strategic focus for many organizations is shifting from simple adoption toward leveraging the cloud for comprehensive digital transformation. Over 90% of enterprises have already implemented cloud technologies in some form, which underscores the widespread recognition of its foundational benefits, such as scalability, flexibility, and cost-efficiency. However, the industry is now moving beyond these basic advantages, as businesses seek to use the cloud to foster innovation, enhance customer experience, and unlock new business models. In this context, a single cloud provider is often insufficient to meet the diverse and needs of enterprises. Consequently, many organizations are adopting a multi-cloud strategy, where they utilize a combination of cloud services from various providers to maximize flexibility, performance, and resilience. This shift reflects a growing recognition that no single cloud solution can fully address the complex requirements of modern digital enterprises [1].

The term "multi-cloud" refers to the strategic use of more than one cloud computing provider to deliver a wide array of IT services. This deployment model allows organizations to distribute workloads across multiple cloud environments, which may include public, private, and hybrid clouds, as well as a range of cloud vendors, regions, and availability zones [2]. The architecture of a multi-cloud environment is designed to support the delivery of services across various cloud platforms and regions. This architecture is typically structured into three primary layers: *Foundational Resources, Workload Management, and Service Consumption*.

The foundational resources form the infrastructure base for all workloads deployed in a multi-cloud environment. These resources include compute, storage, networking, and security components. Compute resources may encompass virtual machines (VMs), containers, and serverless computing, while storage resources range from block and object storage to databases. Networking capabilities include the management of network traffic, firewalls, and virtual private clouds (VPCs), which are essential for ensuring connectivity and security across multiple cloud platforms [3].

Security is a critical component at this layer, as organizations must ensure that their data and workloads are protected across diverse cloud environments. This requires a unified security framework capable of managing encryption, identity and access management (IAM), compliance, and threat detection

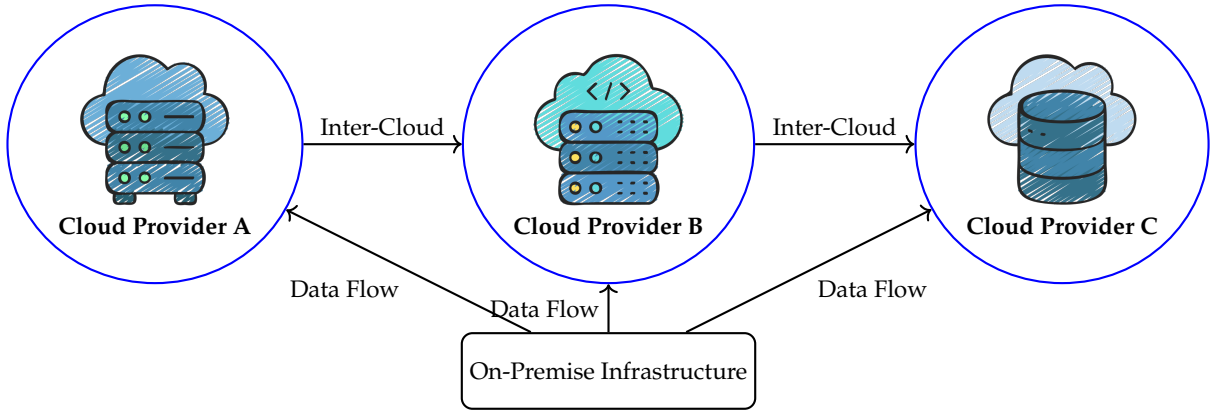

Fig. 1. A simple Multi-Cloud Architecture

No.	Challenge Addressed by Multi-Cloud Computing
1	Handling surges in service or resource demands by utilizing external resources as needed.
2	Optimizing costs or enhancing service quality.
3	Responding to provider offer changes.
4	Adhering to new location-based or legal constraints.
5	Guaranteeing high availability of resources and services.
6	Avoiding reliance on a single external provider.
7	Providing backups to manage disasters or planned downtimes.
8	Serving as an intermediary.
9	Expanding in-house cloud services or resources through agreements with other providers.
10	Leveraging unique services not available elsewhere.

Table 1. Challenges Addressed by Multi-Cloud Computing

No.	Key Factors Driving Multi-Cloud Adoption
1	Enhancing workload performance by allowing enterprises to allocate different workloads to the most suitable cloud environments. Sensitive workloads may be hosted internally, while public or hosted clouds serve other purposes.
2	Preventing vendor lock-in by promoting vendor diversification, enabling businesses to select the most appropriate platforms and easily switch between cloud environments.
3	Reducing the risk of service disruptions by distributing workloads across multiple clouds, minimizing the impact of a failure in any single environment.
4	Improving security through additional protections offered by multiple cloud providers, in case of server disruptions.
5	Enhancing negotiating power by giving enterprises the flexibility to move workloads between clouds, allowing them to secure more favorable pricing.
6	Supporting mergers and acquisitions, as companies often retain multiple cloud platforms following transactions instead of consolidating onto a single platform.

Table 2. Key Factors Driving Multi-Cloud Adoption

across multiple clouds. The complexity of securing a multi-cloud architecture is compounded by the differences in security implementations between cloud providers. Therefore, a comprehensive approach that addresses multi-cloud security challenges is essential.

The second layer of the multi-cloud architecture is workload management. This layer deals with the orchestration, deployment, and lifecycle management of workloads across different cloud environments. Workloads can vary widely in structure and requirements, ranging from traditional monolithic applications to microservices-based architectures. The workload management layer ensures that these workloads can be seamlessly deployed and managed across different cloud platforms without requiring extensive reconfiguration or refactoring.

Technologies such as Kubernetes, OpenStack, and OpenShift play a pivotal role in managing containerized workloads, offering an abstraction layer that allows organizations to deploy applications across multiple cloud environments. By leveraging these container orchestration platforms, enterprises can standardize their deployment and scaling processes, reducing the complexity associated with managing workloads across different clouds.

For applications that do not rely on containerization, traditional virtualization platforms such as VMware or Hyper-V can be employed to manage virtual machines across cloud environments. Moreover, serverless computing, which abstracts the underlying infrastructure, enables enterprises to run event-driven applications that scale automatically across multiple clouds without the need for manual intervention [4].

A key component of the workload management layer is the application lifecycle management framework. This framework facilitates the continuous integration and deployment (CI/CD) of applications, enabling organizations to automate software delivery pipelines and ensure consistent updates across cloud platforms. It also provides visibility into workload performance and availability, allowing IT teams to monitor and optimize workloads based on real-time data. The service consumption layer is where end-users interact with cloud-based applications and services. This layer abstracts the complexities of the underlying infrastructure and workload management, providing a unified interface for users to consume services. The goal of

this layer is to decouple the infrastructure from the services consumed by applications, allowing users to access services without needing to understand the intricacies of how those services are delivered or managed.

In a multi-cloud environment, this layer enables organizations to present a unified front to users, regardless of the underlying cloud infrastructure. This abstraction allows enterprises to offer consistent service levels, application performance, and user experience across different cloud environments, even when those services are spread across multiple regions or cloud vendors [5].

Furthermore, the service consumption layer is closely tied to cloud management platforms (CMPs), which provide centralized control over multiple cloud environments. CMPs allow organizations to manage costs, monitor performance, enforce security policies, and ensure compliance across their multi-cloud environments. By leveraging CMPs, enterprises can gain greater visibility into their cloud usage and costs, helping them optimize resource allocation and reduce waste [6].

A. Types

Multi-cloud architectures can be broadly categorized into two types: composite architecture and redundant architecture, each serving distinct organizational needs based on performance, availability, and resilience requirements. Composite architecture distributes an application portfolio across two or more cloud service providers (CSPs), allowing organizations to leverage the unique strengths of each provider to optimize performance and cost. For instance, an enterprise might run computationally intensive tasks on a cloud provider that excels in high-performance computing while using a different provider with cost-effective storage for data management. This architecture is beneficial when performance optimization is the primary consideration, as it allows organizations to strategically allocate workloads based on the specific advantages of different cloud environments [7]. Applications are not duplicated but instead divided across platforms, utilizing the best offerings of each provider to enhance efficiency and service delivery.

On the other hand, redundant architecture focuses on availability and resilience by deploying multiple instances of the same application across two or more CSPs. This model is essential for mission-critical systems where ensuring uptime and mitigating the risk of failure is paramount. In this architecture, if one cloud fails, another cloud instance seamlessly takes over, ensuring continuous service. Unlike composite architecture, which distributes components for performance, redundant architecture provides a failover mechanism that prioritizes application availability. However, the deployment of the same application on different clouds does not necessarily involve an exact replication. For example, the application running on Cloud A might differ slightly from its instance on Cloud B due to variations in configurations or infrastructure, though both serve the same purpose [1].

Redundant multi-cloud deployments can be further divided into two models: continuously replicated and one-time placement. In continuously replicated deployments, the same application runs concurrently on two or more Infrastructure as a Service (IaaS) CSPs, ensuring real-time synchronization between the instances. This method guarantees that, in the event of a failure in one cloud, the application can immediately switch to an active instance on another cloud without downtime. While this model offers the highest level of resilience, it is resource-intensive, requiring substantial infrastructure to maintain simultaneous instances of the application. In contrast, the one-time placement model involves hosting the application on a single CSP at any given time, with the ability to shift it to another provider if necessary. This model is less demanding in terms of resources since it does not require continuous replication across multiple clouds. However, it may result in some downtime during the failover process, as the application needs to be activated or migrated to another cloud in case of failure.

B. Key Reasons Multi-Cloud Is Gaining Traction

Multi-cloud computing addresses several critical challenges faced by enterprises in managing their IT infrastructure, while offering compelling advantages that explain its growing adoption. One of the primary issues resolved by multi-cloud architectures is the ability to handle peaks in resource demand. Enterprises often experience unpredictable surges in service requests or resource needs, and multi-cloud computing enables them to dynamically scale by tapping into external cloud resources on demand, avoiding the need for costly, over-provisioned infrastructure. This flexibility ensures that capacity is available when required, without incurring unnecessary costs during periods of lower demand [8].

Another significant challenge addressed by multi-cloud is cost optimization and service quality improvement. Different cloud service providers (CSPs) offer a wide variety of pricing models and service levels, which enables organizations to select the most cost-effective solution for their specific needs. By adopting a multi-cloud approach, enterprises can tailor their cloud usage to balance costs and performance, choosing the best provider for each workload or region.

A multi-cloud strategy also allows enterprises to react to changes in cloud provider offerings. As CSPs continuously update their services, pricing, and features, businesses need to maintain the agility to switch providers or reallocate workloads to take advantage of better pricing or new technologies. Regulatory compliance and geographic constraints are another key concern that multi-cloud architectures address. Many organizations, those in industries such as finance and healthcare, must comply with stringent data residency regulations. Multi-cloud setups enable companies to store and process data in specific locations required by law while leveraging global cloud infrastructure to meet other operational needs.

Ensuring high availability and resilience of services is one of the most important challenges in cloud computing. A multi-cloud architecture mitigates the risk of service disruption by distributing workloads across multiple cloud providers. In the event of an outage or technical failure at one provider, workloads can fail over to another cloud environment, ensuring business continuity. This redundancy ensures that critical services remain operational, even in the face of unforeseen disruptions.

Closely tied to high availability is the goal of avoiding dependence on a single provider, often referred to as vendor lock-in. Many enterprises are wary of becoming too reliant on a single cloud vendor, as this can limit flexibility and negotiating power. Multi-cloud strategies allow businesses to diversify their cloud usage, avoiding the constraints of a single-provider model and enabling smoother transitions between cloud platforms.

Multi-cloud computing also enhances disaster recovery and backup capabilities. By replicating workloads and data across multiple cloud environments, organizations can safeguard against data loss and minimize downtime. In case of failures, such as data center outages, enterprises can quickly recover from a backup hosted on a different provider, ensuring continuity of

operations. Additionally, multi-cloud solutions can accommodate scheduled maintenance periods by shifting workloads to other providers, minimizing service disruptions.

Enterprises also use multi-cloud architectures to act as intermediaries between CSPs or to enhance their own cloud offerings through strategic partnerships. This allows them to provide a broader range of services to their customers without needing to invest directly in new infrastructure. Similarly, businesses may consume specific services from multiple providers for their unique attributes—for instance, using one provider for machine learning capabilities and another for data storage—ensuring they can access the best-in-class services for each particular use case.

Several key factors are driving the widespread adoption of multi-cloud strategies. One of the most compelling reasons is the ability to enhance workload performance. Not all workloads perform equally across different cloud environments, and many enterprises find it beneficial to keep sensitive or mission-critical workloads on internal private clouds for better control and security while utilizing public clouds for more scalable, non-sensitive tasks. Multi-cloud strategies enable organizations to align their workloads with the best platform for each specific need, optimizing for performance, security, and cost efficiency [9].

Another important advantage is the ability to avoid vendor lock-in, which is a growing concern for many enterprises. By diversifying their cloud service providers, businesses prevent over-reliance on a single vendor, which can be restrictive and limit flexibility. Multi-cloud strategies enable organizations to adopt a more competitive stance by choosing the best cloud platforms for their needs at any given time, ensuring they can move between vendors or adopt new technologies as they emerge.

Reducing the risk of service disruption is another critical factor driving multi-cloud adoption. Distributing workloads across multiple clouds significantly reduces the chances of a total outage, as the failure of one cloud provider can be mitigated by switching to another. A well-orchestrated multi-cloud approach ensures that services remain available even if one provider experiences a failure. Moreover, each cloud provider typically implements its own security measures, and the combination of different security frameworks can enhance overall system protection, further reducing the risk of data breaches or downtime.

A stronger negotiating position is another key reason multi-cloud is gaining traction. Cloud providers often offer discounts or other incentives to attract or retain customers, and enterprises that can move workloads between providers have more leverage in negotiations. The ability to switch providers or use multiple clouds ensures that companies can obtain better commercial terms, such as lower prices or improved service-level agreements (SLAs), by taking advantage of competitive pressures in the cloud market.

Mergers and acquisitions often push enterprises toward adopting a multi-cloud model. When two companies merge, they typically bring with them different cloud infrastructures. Rather than undergoing a complex and costly process of consolidating under a single provider, many organizations opt to maintain a multi-cloud strategy, allowing them to leverage the existing cloud agreements and infrastructure from both parties. This approach reduces the need for immediate restructuring and enables the enterprise to benefit from the combined strengths of multiple cloud platforms.

2. QUALITY OF SERVICE (QOS) IN MULTI-CLOUD ENVIRONMENTS

Quality of Service (QoS) in cloud computing refers to how well a system performs from the perspective of its users, measured by factors like latency, availability, throughput, fault tolerance, and load balancing. In multi-cloud environments, which use multiple cloud providers, maintaining QoS across different infrastructures is essential to meet Service Level Agreements (SLAs) and keep users satisfied. However, the variety of services and infrastructures offered by different providers adds complexity. Each provider may have specific strengths in areas like performance, resource availability, or pricing, making it harder to optimize QoS consistently. For example, one provider might offer excellent data storage, while another might have better computational resources or perform more effectively in certain regions. As a result, workload distribution, resource allocation, and traffic management need to be managed in real-time to keep QoS at the desired level.

Latency is the delay users experience when accessing services. It can vary significantly depending on data center location, network routing, and the provider's infrastructure. In a multi-cloud setup, managing latency becomes more challenging as providers have different geographic locations and network capabilities. Strategically distributing workloads across providers and using content delivery networks (CDNs) can help reduce latency by bringing data closer to end-users. Availability, the measure of how often cloud services are accessible, can be improved in multi-cloud environments through redundancy and failover strategies. If one provider goes down, another can take over, ensuring continuous service even during outages.

Throughput refers to the amount of data processed and transmitted, is influenced by the capacity of the provider's infrastructure and the type of traffic. Spreading workloads across multiple platforms helps maintain steady throughput and prevent slowdowns. Fault tolerance, the ability to keep running despite hardware or software failures, is important in multi-cloud setups. Although providers offer fault tolerance options, coordinating across platforms requires careful planning since recovery mechanisms may vary.

Load balancingt helps prevent any one resource from becoming overwhelmed. In multi-cloud environments, load balancing must account for differences in capacity, cost, and performance between providers to ensure efficient use of resources and smooth operation. Ongoing monitoring and the ability to adjust resource allocation quickly are important to keep QoS stable as workloads change. The heterogeneity of services and infrastructures among cloud vendors complicates this task. Different providers may offer distinct advantages in terms of pricing, resource avail- ability, or performance in specific regions. For example, one provider may excel in data storage while another may offer superior computational resources. Optimizing QoS in such a sce- nario involves real-time decisions about workload distribution, resource allocation, and traffic management

3. AI-DRIVEN APPROACHES FOR QOS OPTIMIZATION

A. Machine Learning for Dynamic Resource Allocation

Machine learning (ML) has become an essential tool for managing dynamic resource allocation in multi-cloud environments. In such settings, effective resource allocation is crucial for maintaining the Quality of Service (QoS) required by applications and end-users. Traditional resource management techniques

QoS Parameter	Description	Importance in Multi-Cloud Environments
Latency	Time delay between a request and its response.	Reducing latency is critical to ensure fast data processing and user responsiveness, especially across geographically distributed clouds.
Availability	Percentage of time a service remains operational.	High availability ensures continuous access by distributing services across multiple cloud providers.
Throughput	Amount of data processed over a period of time.	Optimizing throughput is key to maintaining efficient data flows between cloud services, avoiding performance bottlenecks.
Fault Tolerance	Ability to continue operations despite component failures.	Fault tolerance is improved by using multiple cloud providers, which reduces the risk of complete service failure.
Load Balancing	Distributes workloads across multiple resources.	Effective load balancing prevents resource overload and maintains stable performance across clouds.

Table 3. QoS Parameters in Multi-Cloud Environments

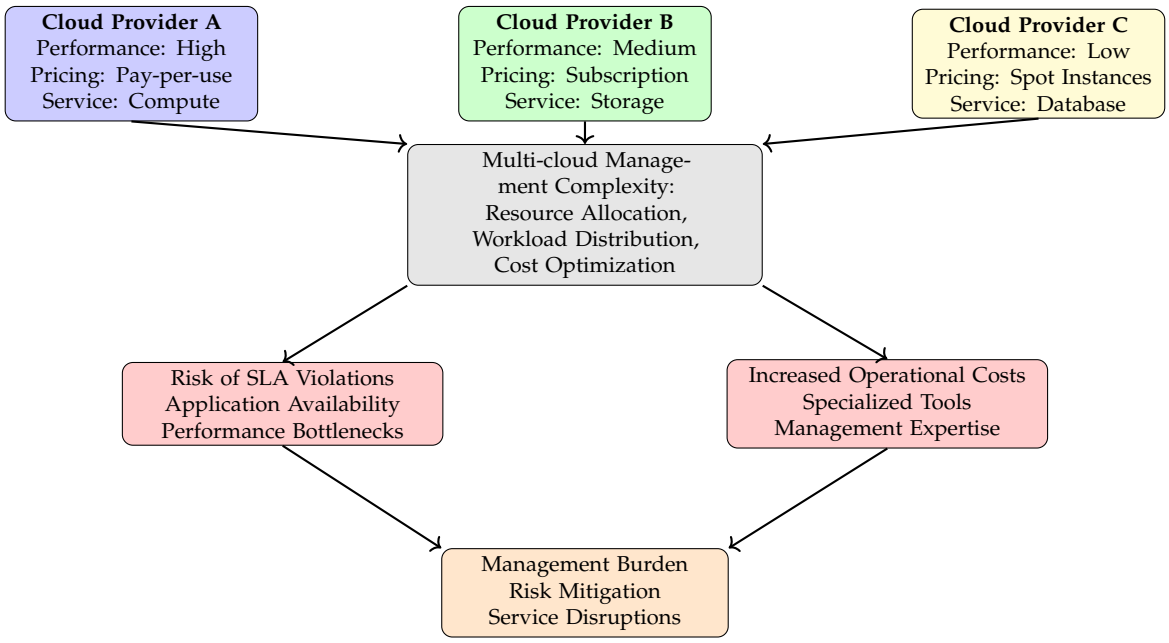

Fig. 2. Illustration of the heterogeneous nature of cloud providers and the resulting management complexity in multi-cloud environments.

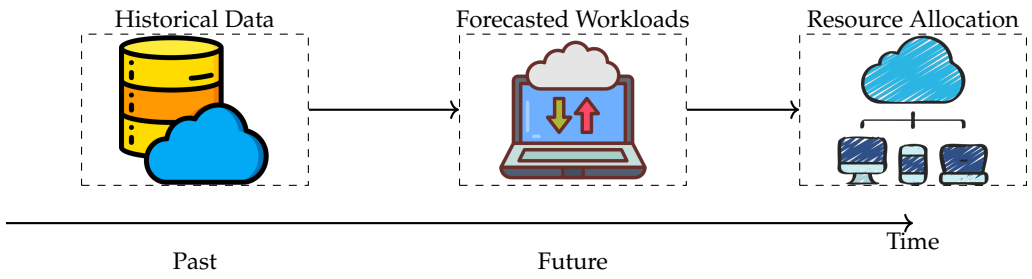

Fig. 3. AI models forecasting future workloads based on historical data, enabling preemptive resource allocation.

often rely on static thresholds or rule-based systems, which struggle to keep up with the complexity and dynamism inherent in multi-cloud architectures. By leveraging ML, predictive models can analyze patterns in resource utilization, workload

demands, and network performance to make informed decisions about resource allocation. These models can forecast future resource needs and dynamically adjust allocations, which minimizes the risks of resource bottlenecks or over-provisioning. This approach ensures a more efficient utilization of resources while maintaining the desired QoS, even in fluctuating cloud environments [10] [11].

Algorithm 1. Supervised Learning for QoS Prediction in Multi-Cloud

Input: Labeled dataset $D = \{(X_i, y_i)\}_{i=1}^{n}$ with X_i as resource utilization, network traffic, and performance metrics; target QoS level y_i.
Output: Predicted QoS and resource allocation decision.
foreach *new workload* W_j **do**
  Extract features X_j from the workload W_j Predict QoS $\hat{y}_j = f(X_j)$ using trained model f **if** $\hat{y}_j <$ *QoS threshold* **then**
    Dynamically allocate more resources to workload W_j
  end
  else
    Maintain current resource allocation
  end
end

Algorithm 2. Reinforcement Learning for Dynamic Resource Allocation in Multi-Cloud

Input: State space S, action space A, reward function R, and environment E representing multi-cloud conditions.
Output: Optimized policy π for resource allocation.
Initialize policy π **while** *cloud operations running* **do**
  Observe current state $s_t \in S$ Select action $a_t \in A$ using policy $\pi(a_t|s_t)$ (e.g., scaling resources, rerouting traffic) Execute action a_t in environment E Observe reward $r_t = R(s_t, a_t)$ and next state s_{t+1} Update policy π using reward r_t and next state s_{t+1} **if** *reward* r_t *improves QoS* **then**
    Continue current actions
  end
  else
    Adjust resource allocation or network routing
  end
end

Algorithm 3. Unsupervised Learning for Anomaly Detection in Multi-Cloud

Input: Unlabeled dataset $D = \{X_i\}_{i=1}^{n}$ with resource utilization and performance metrics.
Output: Detected anomalies in resource utilization or network performance.
Apply clustering algorithm (e.g., k-means) on dataset D **foreach** *cluster* **do**
  Calculate centroid c_k for each cluster **foreach** *new data point* X_j **do**
    Compute distance $d_j = \|X_j - c_k\|$ **if** $d_j >$ *threshold* **then**
      Mark X_j as anomaly (e.g., latency spike, resource imbalance) Trigger corrective action to rebalance resources
    end
  end
end

Supervised learning is one of the most straightforward approaches to integrating ML for dynamic resource allocation. This method involves training models on historical datasets that contain labeled data, such as resource utilization statistics, network traffic patterns, and QoS performance metrics. By analyzing these datasets, supervised learning models can predict future QoS performance based on current and anticipated conditions.

The key advantage of supervised learning in this context is its ability to generalize patterns from historical data, providing a reliable basis for forecasting resource needs. For example, workloads in cloud environments often exhibit diurnal or weekly usage patterns that can be predicted with high accuracy. By learning these patterns, supervised models can anticipate spikes or drops in demand and allocate resources accordingly, thus preventing resource bottlenecks that could degrade QoS. Additionally, these models can predict underutilization and allow cloud operators to scale down resources, avoiding unnecessary costs associated with over-provisioning.

Training supervised models for cloud resource allocation requires large and representative datasets. These datasets should contain a diverse set of conditions, including different levels of workload intensity, network congestion, and service-level agreement (SLA) constraints. Given the variability in cloud environments, ensuring that the model has been exposed to an appropriate range of scenarios is critical for its generalizability. After training, the model can be deployed in real-time environments, where it continuously monitors key performance indicators (KPIs) such as CPU and memory utilization, network latency, and response times. Based on its predictions, the system can proactively adjust resource allocations before performance issues arise, ensuring that QoS targets are consistently met.

While supervised learning is effective for making predictions based on historical data, reinforcement learning (RL) offers a more dynamic and adaptive approach. In a multi-cloud environment where conditions can change rapidly due to variable workloads, shifting user demands, or failures in individual cloud components, RL models can learn to optimize resource allocation in real time.

Reinforcement learning operates based on a reward system, where the model learns by interacting with the environment and receiving feedback on the quality of its actions. In the context of cloud resource management, the actions can include scaling resources up or down, rerouting network traffic, or adjusting load-balancing algorithms. The reward function, which guides the learning process, is typically designed to maximize QoS while minimizing costs or energy consumption. Over time, the RL model develops a policy that dictates the optimal actions to take in response to varying environmental conditions.

One of the primary advantages of RL in multi-cloud environments is its ability to adapt to unforeseen changes. Unlike supervised learning, which is dependent on historical data, RL continuously learns from its interactions with the system. For instance, if a particular cloud provider experiences a sudden drop in network performance, the RL model can quickly reroute traffic to a different provider with better QoS metrics [12]. This capability is used in multi-cloud architectures where conditions are inherently dynamic and unpredictable.

Moreover, RL models can incorporate feedback loops to refine their performance over time. By continuously adjusting their policies based on real-time performance data, these models can achieve near-optimal resource allocation strategies. This adaptive approach also helps in balancing trade-offs between competing objectives, such as minimizing latency while reduc-

Learning Technique	Application in Multi-Cloud QoS Optimization
Supervised Learning	Predicts QoS performance based on historical data. Models trained on labeled datasets (e.g., resource utilization, network traffic, performance metrics) can anticipate future QoS needs and adjust resource allocations accordingly.
Reinforcement Learning	AI systems learn from cloud environment interactions, continuously adjusting actions (e.g., scaling resources, rerouting traffic) to maximize QoS. useful in dynamic multi-cloud environments, where conditions change rapidly.
Unsupervised Learning	Clustering and anomaly detection algorithms identify abnormal behaviors in cloud performance. Detects issues such as latency spikes or resource imbalances, enabling systems to react before SLA violations occur.

Table 4. Applications of Machine Learning Techniques in Multi-Cloud QoS Optimization

ing operational costs. For example, an RL model may decide to allocate additional resources during peak usage hours to meet QoS requirements and then scale back during off-peak hours to conserve energy and reduce costs [13].

While supervised and reinforcement learning methods focus on optimizing resource allocation and QoS, unsupervised learning techniques play a complementary role by enhancing system reliability and resilience [14]. Specifically, clustering and anomaly detection algorithms can be used to identify abnormal behaviors in cloud performance, such as latency spikes, resource imbalances, or unexpected workload surges. By detecting these anomalies early, unsupervised learning models can help prevent QoS degradation and SLA violations.

In a multi-cloud environment, performance issues can stem from a wide range of factors, including hardware failures, software bugs, network congestion, or misconfigured resources. Since these issues are often difficult to predict and may not follow historical patterns, supervised learning approaches may struggle to detect them. Unsupervised learning, on the other hand, excels at identifying deviations from the norm without requiring labeled training data.

Clustering algorithms, such as k-means or DBSCAN, can group similar resource utilization patterns or performance metrics into clusters. By monitoring these clusters over time, the system can establish a baseline for normal behavior. When new data points fall outside of these established clusters, the system can flag them as potential anomalies. For instance, a sudden spike in CPU utilization that does not align with typical workload patterns may indicate a potential issue, such as a distributed denial-of-service (DDoS) attack or an inefficient resource configuration.

Anomaly detection models can also integrate real-time monitoring data to provide immediate alerts when performance deviates from expected levels. This proactive approach allows cloud operators to take corrective actions, such as redistributing resources or adjusting network configurations, before performance issues escalate and affect end-users. For example, if the anomaly detection model identifies a sudden increase in network latency, it could trigger a reallocation of traffic to a less congested path, preventing service disruptions.

Another important use case for unsupervised learning in multi-cloud environments is fault detection. Cloud systems often rely on distributed architectures with multiple points of failure, and detecting faults before they lead to widespread outages is critical for maintaining high availability. By continuously analyzing performance data, unsupervised models can identify subtle signs of degradation, such as increased error rates or fluctuations in response times, that may indicate an impending failure. Early detection allows cloud operators to implement failover strategies or initiate maintenance before the fault impacts QoS.

The integration of machine learning into dynamic resource allocation frameworks for multi-cloud environments offers a robust solution to the challenges of managing complex, distributed architectures. Each of the learning techniques—supervised, reinforcement, and unsupervised—brings distinct advantages that, when combined, can provide a comprehensive approach to resource management.

For instance, a hybrid system could use supervised learning to predict resource demands based on historical data, reinforcement learning to adjust allocations in real-time based on current conditions, and unsupervised learning to monitor for anomalies and ensure system reliability. This multi-faceted approach allows for both proactive and reactive management of cloud resources, enhancing the overall performance and resilience of the cloud environment.

Moreover, machine learning models can be integrated with cloud orchestration tools, such as Kubernetes or OpenStack, to automate the resource management process. These tools allow cloud operators to define policies for resource allocation, such as auto-scaling rules, which can be augmented by machine learning algorithms. For example, an RL model could work in tandem with Kubernetes' auto-scaler to optimize the scaling of containerized applications based on real-time QoS metrics [15]. Similarly, unsupervised learning models could be used to detect and mitigate performance issues before they lead to SLA violations.

Furthermore, the use of machine learning for dynamic resource allocation in multi-cloud environments also addresses the challenge of heterogeneity. Multi-cloud environments often involve a mix of public, private, and hybrid cloud infrastructures, each with its own set of performance characteristics and pricing models. Machine learning models can optimize resource allocation across these different environments by taking into account factors such as workload performance requirements, cost constraints, and network latency. For instance, a supervised learning model could predict which cloud provider will offer the best performance for a specific workload at a given time, while an RL model could dynamically allocate resources to the most cost-effective provider based on current conditions.

B. Deep Learning for Predictive Analytics and Automation

Deep learning is proving to be a highly effective approach for predictive analytics and automation in multi-cloud management.

Its ability to process large datasets and make complex decisions in real-time is used in optimizing resource allocation and maintaining Quality of Service (QoS). In this context, three specific deep learning techniques—neural networks, autoencoders, and reinforcement learning—stand out for their practical applications [16].

Algorithm 4. Neural Networks for Workload Prediction in Multi-Cloud

Input: Historical dataset $D = \{X_i\}_{i=1}^{n}$ containing workload and resource utilization metrics.
Output: Predicted future workload $\hat{W}_t$.
Train neural network NN on dataset D **foreach** *time step t* **do**
  Input current system metrics X_t into the neural network NN Predict future workload $\hat{W}_t = NN(X_t)$ **if** $\hat{W}_t > threshold$ **then**
    Scale resources to accommodate predicted workload
  end
  else
    Maintain current resource allocation
  end
end

Algorithm 5. Autoencoders for Anomaly Detection in Multi-Cloud

Input: Dataset $D = \{X_i\}_{i=1}^{n}$ of performance metrics under normal conditions.
Output: Detected anomalies in system performance.
Train autoencoder AE on dataset D to learn compressed representation of normal conditions **foreach** *new data point* X_j **do**
  Reconstruct data point $\hat{X}_j = AE(X_j)$ Compute reconstruction error $e_j = \|X_j - \hat{X}_j\|$ **if** $e_j > anomaly\ threshold$ **then**
    Flag X_j as an anomaly Trigger corrective action to prevent QoS degradation
  end
end

One of the most useful applications of deep learning in multi-cloud environments is workload prediction. Neural networks, deep learning architectures, excel at identifying complex patterns in large-scale datasets. This makes them highly suited for predicting future workloads with greater accuracy than traditional models. Accurate workload prediction is essential for dynamic resource allocation because it allows cloud systems to anticipate demand and adjust resources proactively.

Neural networks are able to capture the non-linear relationships between variables, which are often present in cloud environments where resource needs fluctuate based on factors like user demand, network congestion, or application load. By learning from historical workload data, these models can predict future spikes or dips in resource usage, allowing cloud systems to automatically scale resources up or down as needed. This leads to more efficient resource utilization, reducing the risk of over-provisioning or resource shortages, which can negatively impact QoS.

Moreover, neural networks can be retrained regularly as new data becomes available, ensuring that their predictions remain accurate even as workload patterns change over time. This ability to adapt to changing conditions makes deep learning models well-suited for the dynamic nature of multi-cloud environments.

In multi-cloud systems, maintaining high levels of performance and reliability requires early detection of anomalies that could affect QoS. Autoencoders, a form of unsupervised deep learning, are well-suited for this task. These models are designed to learn a compressed representation of normal operating conditions in the cloud environment. By comparing new data to this learned baseline, autoencoders can flag deviations that may indicate potential issues, such as performance bottlenecks, network failures, or security breaches.

The key strength of autoencoders lies in their ability to detect subtle anomalies that might go unnoticed by simpler threshold-based systems. For example, a slight but consistent increase in response times or a small fluctuation in resource utilization might not trigger traditional alarms, but could be an early sign of a developing problem. Autoencoders, trained on the normal behavior of the system, can detect such anomalies at an early stage, allowing cloud operators to take preemptive action before these issues escalate into significant outages or SLA violations.

Autoencoders can also be used in conjunction with other monitoring tools to provide a more comprehensive view of system health. By integrating anomaly detection into the broader cloud management framework, autoencoders can help ensure that performance issues are addressed promptly, minimizing their impact on end users.

Managing SLAs in a multi-cloud environment is a complex task, as it requires balancing multiple, often competing, objectives such as minimizing costs while maintaining high levels of performance. Deep reinforcement learning (RL) offers a powerful solution for automating this process. In this approach, an RL model is trained to make resource allocation decisions that maximize long-term QoS while minimizing costs and avoiding SLA violations [17].

Reinforcement learning is useful in multi-cloud environments because it allows for continuous learning and adaptation. The RL model learns by interacting with the environment—scaling resources, rerouting traffic, or switching between cloud providers—and receiving feedback based on whether these actions improve QoS or reduce costs. Over time, the model develops an optimal policy that allows it to make more informed decisions as conditions change [12].

For SLA management, deep reinforcement learning models can be designed to prioritize actions that prevent violations. For example, the model could decide to allocate more resources to a critical application during periods of high demand, even if this increases short-term costs, in order to avoid a costly SLA breach. Similarly, the model could optimize resource allocation across different cloud providers based on factors like performance metrics, costs, and network latency, ensuring that SLAs are consistently met while keeping operational expenses as low as possible.

The integration of deep learning into multi-cloud environments offers a substantial advantage in terms of automation and decision-making. By combining neural networks for workload prediction, autoencoders for anomaly detection, and reinforcement learning for SLA management, cloud systems can become more adaptive and resilient. This combination of techniques allows for both proactive and reactive management strategies, ensuring that resources are allocated efficiently and potential issues are addressed before they impact service delivery.

Additionally, deep learning models can be incorporated into existing cloud management platforms and tools, such as Ku-

AI Technique	Application in Multi-Cloud Environments
Neural Networks for Workload Prediction	Deep learning models process large-scale datasets to predict future workloads with high accuracy. These predictions optimize real-time resource allocation by identifying intricate patterns often missed by simpler models.
Autoencoders for Anomaly Detection	Autoencoders, a type of unsupervised deep learning model, detect performance anomalies in multi-cloud environments by learning compressed representations of normal operating conditions and flagging deviations that could affect QoS.
Reinforcement Learning for SLA Management	Deep reinforcement learning autonomously manages resources across multiple cloud providers, making decisions that optimize long-term QoS while minimizing costs, thereby reducing SLA violations.

Table 5. AI Techniques for Multi-Cloud QoS Optimization and SLA Management

bernetes or Terraform, to further automate the resource management process. These tools can integrate the predictive capabilities of neural networks with the adaptive optimization of reinforcement learning, allowing for real-time adjustments in resource allocation based on predicted workload patterns or current system performance.

C. Reinforcement Learning for SLA Violation Reduction

SLAs typically define performance metrics such as uptime, latency, or throughput, and violations of these agreements can lead to financial penalties and degraded user experiences. Reinforcement learning provides a robust solution by enabling cloud systems to learn optimal policies for resource management, adapting to fluctuating conditions in real-time to prevent breaches.

Algorithm 6. MDP for Resource Allocation in Multi-Cloud

Input: State space S, action space A, transition probabilities $P(s'|s,a)$, reward function $R(s,a)$, discount factor γ.
Output: Optimal policy π^* for resource allocation.
Initialize value function $V(s)$ for all $s \in S$ **while** *not converged* **do**
  foreach *state* $s \in S$ **do**
    foreach *action* $a \in A$ **do**
      Compute expected return $Q(s,a) = \sum_{s'} P(s'|s,a)\left[R(s,a) + \gamma V(s')\right]$
    end
    Update $V(s) = \max_a Q(s,a)$
  end
end
Obtain optimal policy $\pi^*(s) = \arg\max_a Q(s,a)$ for each state s

Algorithm 7. Q-Learning for Multi-Cloud QoS Management

Input: State space S, action space A, learning rate α, discount factor γ, and exploration parameter ϵ.
Output: Learned Q-values $Q(s,a)$ and optimal policy π.
Initialize Q-table $Q(s,a)$ for all $s \in S$ and $a \in A$ **while** *cloud operations are running* **do**
  Observe current state $s_t \in S$ **if** *random number* $< \epsilon$ **then**
    Choose random action $a_t \in A$ (exploration)
  end
  else
    Choose action $a_t = \arg\max_a Q(s_t,a)$ (exploitation)
  end
  Execute action a_t and observe reward r_t and next state s_{t+1}
  Update Q-value:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left[r_t + \gamma \max_{a'} Q(s_{t+1},a') - Q(s_t,a_t)\right]$$

  Update policy $\pi(s_t) = \arg\max_a Q(s_t,a)$
end

Markov Decision Processes (MDPs) are fundamental to reinforcement learning frameworks and are useful for addressing SLA violations in multi-cloud environments. In this context, an MDP provides a structured approach to decision-making, allowing systems to manage resources dynamically while minimizing the risk of SLA breaches. The process involves defining the current state of the cloud environment, which includes variables such as resource usage, network performance, and proximity to SLA thresholds. Actions represent the decisions available to the system, such as allocating additional resources, redistributing workloads across different cloud providers, or adjusting traffic routing strategies. The reward mechanism within the MDP helps guide decision-making, where the system receives positive rewards for actions that maintain or improve QoS and negative rewards for actions that lead to SLA violations. Over time, by learning the optimal policy, the system can determine the best sequence of actions to minimize performance issues and reduce the likelihood of SLA breaches. This structured approach is effective in multi-cloud environments, where the complexity and variability of conditions make it difficult to predict performance using static rules. Q-learning, a model-free reinforcement learning algorithm, is well-suited for adaptive SLA management in multi-cloud environments. Unlike traditional methods that require a predefined model of the environment, Q-learning allows systems to learn optimal actions based on experience, without needing an explicit model of the environment's dynamics. This

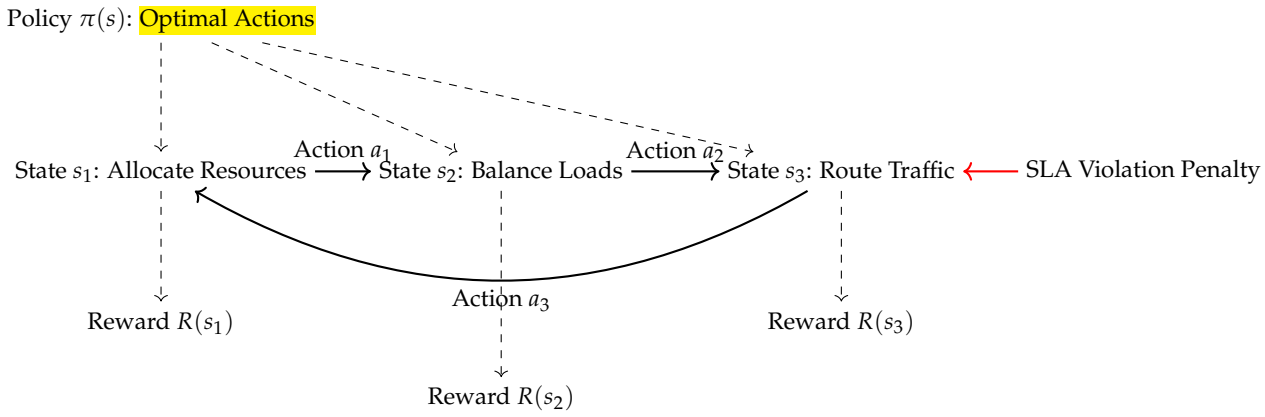

Fig. 4. Simplified MDP for Multi-cloud QoS Management

Reinforcement Learning Technique	Application in Multi-Cloud Environments
Markov Decision Processes (MDPs)	MDPs are used to model decision-making problems in multi-cloud QoS management. They help identify the optimal sequence of actions for resource allocation, load balancing, and traffic routing, aiming to prevent SLA violations.
Q-Learning	Q-learning, a model-free reinforcement learning technique, enables systems to learn the best actions for a given state without needing an explicit model of the environment. It is beneficial in heterogeneous and unpredictable multi-cloud environments.

Table 6. Reinforcement Learning Techniques in Multi-Cloud QoS Management

is especially useful in multi-cloud environments, where the performance characteristics of different cloud providers can vary widely and unpredictably. The diagram in figure 4 illustrates a simplified Markov Decision Process (MDP) model for managing Quality of Service (QoS) in a multi-cloud environment. In this context, states represent different stages of resource allocation and traffic management. For instance, at state s_1, resources are allocated across multiple clouds, followed by load balancing at state s_2, and routing traffic at state s_3. Each transition between states is governed by specific actions, such as reallocating resources, rebalancing loads, or rerouting traffic between clouds. These actions aim to optimize cloud performance and minimize the risk of Service Level Agreement (SLA) violations.

At each state, rewards are associated with the success of the actions taken, which reflect the system's performance in terms of load distribution, latency, and resource utilization. A policy $\pi(s)$ guides decision-making, recommending the optimal action at each state to maximize the cumulative reward. If an incorrect action is chosen, leading to non-optimal routing or load balancing, an SLA violation penalty may occur, as indicated in the diagram.

In Q-learning, the system explores different actions (such as scaling resources or rerouting traffic) in various states (such as high network latency or increased workload demand) and assigns each action-state pair a value, known as the Q-value. Over time, by interacting with the environment and receiving feedback in the form of rewards or penalties, the system learns which actions are most likely to prevent SLA violations in specific states. For example, when a workload begins to exceed a certain threshold that risks breaching an SLA, the Q-learning model might learn to automatically allocate more resources or balance the load across different cloud providers to prevent degradation in performance.

A major advantage of Q-learning is its ability to handle the heterogeneous and constantly changing landscape of multi-cloud environments. Since it does not rely on a static model of the environment, the system can continue to learn and adapt to new conditions, such as sudden spikes in traffic or changes in the performance of individual cloud providers. This flexibility makes Q-learning a powerful tool for managing the complexities of SLA compliance, allowing cloud systems to minimize violations even in the face of unpredictable conditions.

D. Federated Learning for Distributed Cloud Environments

Federated learning has gained significant traction as a decentralized approach to machine learning, for distributed cloud environments. In multi-cloud architectures, where multiple cloud providers manage their own resources and data, federated learning offers a solution that enables collaborative model training without the need to transfer sensitive or private data across different providers. This approach is crucial in scenarios where data privacy and security are critical concerns, such as in industries subject to stringent data protection regulations. By allowing models to be trained locally on each cloud provider's data while aggregating knowledge globally, federated learning enhances both the performance of cloud systems and their adherence to privacy standards.

One of the key advantages of federated learning in multi-cloud environments is its ability to support collaborative model training across multiple cloud providers. In traditional centralized machine learning, data from different sources is aggregated in one location for model training, which can raise concerns about data privacy, especially in sensitive industries like healthcare, finance, or government. Federated learning addresses this issue by allowing each cloud provider to train the model locally on its own dataset, ensuring that sensitive information never

Federated Learning Technique	Application in Multi-Cloud Environments
Collaborative Model Training	Federated learning allows cloud providers to collaboratively train AI models on their local data, ensuring that sensitive information remains within the provider's infrastructure. This decentralized approach improves workload prediction and resource optimization accuracy while maintaining data security.
Privacy-Preserving QoS Optimization	In situations where data privacy regulations restrict sharing user data between cloud providers, federated learning ensures AI-driven QoS optimization continues, maintaining compliance with privacy rules while optimizing service quality.

Table 7. Federated Learning Applications in Multi-Cloud Environments

leaves the provider's infrastructure.

After the local models are trained, the system aggregates the learned parameters (e.g., weights of a neural network) at a central server without transferring the raw data itself. This aggregated model is then updated and distributed back to each provider for further training. Through this iterative process, cloud providers can collaboratively develop highly accurate models for workload prediction, resource optimization, and QoS management, without exposing sensitive user or operational data to external entities. This decentralized model training process helps maintain strong data security, while still benefiting from the collective insights gained across the distributed cloud environment.

For instance, federated learning can be used to train machine learning models that predict future resource needs based on local usage patterns at each cloud provider. By combining insights from different providers without sharing the actual usage data, the model becomes more robust and can account for diverse conditions present in a multi-cloud ecosystem. This leads to more accurate resource allocation, preventing bottlenecks or over-provisioning, and ultimately improving overall system performance.

Federated learning is used in situations where data privacy regulations, such as GDPR or HIPAA, prevent the sharing of user data between cloud providers. These regulations impose strict limitations on how user data can be collected, stored, and transferred, especially across national borders. In multi-cloud environments, where workloads and data may span multiple jurisdictions and providers, complying with these regulations while optimizing QoS can be a major challenge. Federated learning provides a privacy-preserving mechanism for achieving this balance.

In QoS optimization, federated learning enables each cloud provider to train models locally on data that reflects their specific performance metrics and user demands, without exposing this data to other providers. This localized approach ensures compliance with data privacy laws while still allowing cloud systems to leverage the collective knowledge from all providers involved. For example, a federated model can learn to predict latency patterns or optimize traffic routing based on local conditions at each provider, while also incorporating broader trends from the global system.

The decentralized nature of federated learning also reduces the risk of a single point of failure or data breach, which is a concern in centralized machine learning frameworks where all data is pooled in one location. Since no raw data is transferred between providers, the attack surface for potential data breaches is minimized, enhancing the security posture of the entire multi-cloud ecosystem. Additionally, federated learning's focus on privacy ensures that AI-driven QoS optimization continues uninterrupted, even in environments with strict data compliance requirements.

The use of federated learning in multi-cloud environments presents several benefits. First and foremost, it enables collaborative learning across different cloud providers while maintaining strong data privacy and security standards. This approach allows for improved accuracy in predictive analytics and resource management models, as the system benefits from the collective learning of diverse datasets without compromising sensitive information.

Second, federated learning supports compliance with data privacy regulations, which is becoming increasingly important as more industries rely on cloud services for managing critical operations. By training models locally and only sharing model updates, cloud providers can ensure that user data remains secure and compliant with regulatory frameworks, even in highly distributed systems.

One of the primary difficulties is the heterogeneity of the data and infrastructure across different cloud providers. Each provider may have different hardware, network architectures, or data distributions, which can introduce complexity into the model aggregation process. Variations in data quality and consistency across providers may also affect the performance of the federated model, leading to challenges in achieving uniform accuracy. Another challenge lies in communication overhead. Federated learning involves frequent exchanges of model updates between providers and the central server, which can increase network traffic and processing time, especially in large-scale environments. Addressing these challenges will require advances in communication-efficient algorithms and techniques that can handle the diverse and distributed nature of multi-cloud environments.

4. CONCLUSION

The rapid rise of cloud computing has led many businesses to adopt multi-cloud strategies, using multiple cloud service providers (CSPs) to distribute their workloads. This approach offers flexibility, allowing organizations to optimize cost, performance, and reliability by leveraging the strengths of different cloud vendors. For instance, one provider may offer better storage options, while another may excel in computational power. However, managing these diverse platforms poses significant challenges, when it comes to ensuring consistent Quality of Service (QoS) across all providers.

Each cloud vendor has its own pricing, performance metrics, and service agreements, which introduces complexities in maintaining optimal QoS. Issues such as variability in latency, availability, and network performance between providers can

lead to inefficiencies, SLA violations, and increased operational costs. These challenges are exacerbated in dynamic environments where workloads and resource needs fluctuate.

Traditional methods of resource management, which often rely on static rules or manual configurations, struggle to keep up with the dynamic and scalable nature of multi-cloud systems. This is where artificial intelligence (AI) offers a compelling solution. With its ability to analyze patterns, predict outcomes, and make real-time decisions, AI can play a key role in addressing the challenges of managing QoS in multi-cloud environments. Specifically, machine learning (ML) and deep learning (DL) algorithms have the potential to automate resource allocation, predict workload demands, and optimize performance across multiple cloud providers.

This paper explores how AI-driven techniques can be applied to dynamically optimize QoS in multi-cloud environments. The focus is on the application of machine learning and deep learning methods, their role in managing different QoS parameters, and how these techniques can reduce inefficiencies, improve resource utilization, and minimize SLA violations.

Quality of Service (QoS) in cloud computing refers to the overall performance and reliability of a system as experienced by the end-user. In a multi-cloud setting, maintaining consistent QoS is critical to meeting service level agreements (SLAs) and ensuring user satisfaction. The primary QoS parameters in these environments include latency, throughput, availability, fault tolerance, and load balancing. Managing these parameters becomes especially challenging when dealing with the heterogeneity of services and infrastructure across multiple cloud providers.

Each provider may offer distinct benefits—one may have lower latency in specific regions, while another offers more cost-effective storage. This diversity complicates the task of optimizing QoS because it requires careful management of resources across multiple platforms. For example, a system might need to decide in real-time whether to route traffic through a provider offering lower latency or to prioritize cost by using a more affordable but slightly slower service.

Achieving this balance is important in situations where workloads fluctuate or where the system must scale quickly to meet demand. Traditional static methods for resource management struggle with this level of complexity, especially when resources and traffic patterns change dynamically. AI-driven techniques, however, are well-suited for handling such challenges because they can learn from historical data and make real-time decisions that optimize QoS.

Artificial Intelligence has proven to be highly effective in automating complex tasks and improving decision-making in various industries, and cloud computing is no exception. In multi-cloud environments, AI can be used to predict workloads, dynamically allocate resources, and make real-time adjustments to prevent QoS degradation.

Machine learning, a subset of AI, enables systems to learn from historical data and improve their performance over time. This is useful for predicting resource demand and optimizing allocation in a multi-cloud environment. For example, ML models can analyze past usage patterns to forecast future resource needs, allowing systems to allocate just the right amount of resources to meet demand without over-provisioning or under-provisioning.

Deep learning uses neural networks to identify patterns and make complex predictions. In multi-cloud management, deep learning models can process large amounts of data from different cloud providers to make more accurate decisions about resource allocation and performance optimization. For example, deep learning models can predict traffic spikes and allocate additional resources in real time to prevent latency issues.

AI-driven techniques offer various methods to improve QoS in multi-cloud environments. These approaches typically revolve around using machine learning and deep learning algorithms to make real-time decisions that ensure optimal performance and resource efficiency. Machine learning can play a pivotal role in dynamic resource allocation within a multi-cloud environment. By analyzing historical data, ML models can forecast future resource needs and allocate resources dynamically, ensuring that workloads receive the necessary compute power and storage without wasting resources. For example, supervised learning can be used to train models on labeled data sets that include resource usage patterns, network traffic, and performance metrics. These models can then predict future resource needs and automatically adjust allocations in real time. This helps to prevent bottlenecks that might degrade QoS.

Additionally, reinforcement learning can be applied to optimize resource management over time. In this approach, an AI model learns from its interactions with the cloud environment, adjusting its actions (such as scaling resources or rerouting traffic) to maximize QoS. Unsupervised learning, including clustering and anomaly detection, can also be used to identify abnormal patterns in cloud performance. For example, these techniques can detect latency spikes or resource imbalances before they lead to SLA violations, allowing the system to make proactive adjustments.

Deep learning models, neural networks, are powerful tools for predicting future workload demands and automating cloud management tasks. These models are capable of identifying intricate patterns in large data sets, enabling more accurate predictions of future resource needs.

For instance, neural networks can be used to predict traffic patterns or workload demands based on historical data. This allows systems to scale resources up or down as needed, ensuring that performance remains consistent without over-allocating resources.

Autoencoders, a type of unsupervised deep learning model, can be used for anomaly detection in multi-cloud environments. These models learn the normal operating conditions of a system and can identify deviations from the norm, helping to detect potential issues before they impact QoS.

Reinforcement learning, combined with deep learning techniques, can also be used to manage SLA compliance. In this case, an AI system learns to make decisions that maximize long-term QoS while minimizing operational costs, helping to prevent SLA violations and improve overall system efficiency.

Federated learning offers a promising approach for improving QoS in multi-cloud environments, especially when data privacy and security are critical concerns. Unlike traditional machine learning, where data is centralized for model training, federated learning enables AI models to be trained across multiple providers without exchanging sensitive data. This decentralized approach ensures that the learning process respects the privacy and security constraints of different cloud environments.

Federated learning can be useful for optimizing QoS across different cloud providers, as it allows models to be trained on local data at each provider while still benefiting from the collective knowledge of the entire system. This enables more accurate predictions of resource needs and workload distribution without compromising data security or violating privacy regulations.

In scenarios where compliance with data privacy laws is paramount, federated learning can ensure that AI-driven opti-

mizations continue without the need to share sensitive user data between cloud providers. This approach allows for effective QoS management in environments where data cannot be moved freely between providers due to legal or regulatory constraints [18].

The application of AI techniques in multi-cloud environments offers significant potential for optimizing QoS, reducing SLA violations, and improving resource efficiency. By leveraging machine learning, deep learning, and federated learning methods, organizations can automate the management of complex, dynamic cloud environments and ensure that performance remains consistent across multiple providers. While the study focuses on machine learning (ML), deep learning (DL), and reinforcement learning (RL), the scalability of these techniques in real-world multi-cloud scenarios remains uncertain due to the significant computational resources required to process large volumes of heterogeneous data in real time.

Another limitation is the issue of data heterogeneity and interoperability between cloud providers. Different Cloud Service Providers (CSPs) often have incompatible data formats, APIs, and performance metrics, which can complicate the seamless application of AI models across multiple platforms. This can limit the generalizability of the proposed AI models, as they may require customization or additional preprocessing to work effectively with each provider's specific infrastructure.

This study primarily emphasizes the optimization of traditional QoS parameters like latency, throughput, and availability but may not account sufficiently for newer, emerging requirements such as compliance, privacy, and legal constraints, which are increasingly critical in multi-cloud environments. This limitation reduces the relevance of the research for industries dealing with sensitive data or stringent regulatory requirements, such as healthcare or finance.

REFERENCES

1. P. Jamshidi, C. Pahl, and N. C. Mendonça, "Pattern-based multi-cloud architecture migration," Software: Pract. Exp. **47**, 1159–1184 (2017).
2. D. Petcu, "Multi-cloud: expectations and current approaches," in *Proceedings of the 2013 international workshop on Multi-cloud applications and federated clouds,* (2013), pp. 1–6.
3. M. M. Alshammari, A. A. Alwan, A. Nordin, and I. F. Al-Shaikhli, "Disaster recovery in single-cloud and multi-cloud environments: Issues and challenges," in *2017 4th IEEE international conference on engineering technologies and applied sciences (ICETAS),* (IEEE, 2017), pp. 1–7.
4. A. J. Ferrer, D. G. Pérez, and R. S. González, "Multi-cloud platform-as-a-service model, functionalities and approaches," Procedia Comput. Sci. **97**, 63–72 (2016).
5. N. Ferry, F. Chauvel, A. Rossini, *et al.*, "Managing multi-cloud systems with cloudmf," in *Proceedings of the Second Nordic Symposium on Cloud Computing & Internet Technologies,* (2013), pp. 38–45.
6. J. Hong, T. Dreibholz, J. A. Schenkel, and J. A. Hu, "An overview of multi-cloud computing," in *Web, Artificial Intelligence and Network Applications: Proceedings of the Workshops of the 33rd International Conference on Advanced Information Networking and Applications (WAINA-2019) 33,* (Springer, 2019), pp. 1055–1068.
7. M. Abouelyazid, "Forecasting resource usage in cloud environments using temporal convolutional networks," Appl. Res. Artif. Intell. Cloud Comput. **5**, 179–194 (2022).
8. Y. Singh, F. Kandah, and W. Zhang, "A secured cost-effective multi-cloud storage in cloud computing," in *2011 IEEE conference on computer communications workshops (INFOCOM WKSHPS),* (IEEE, 2011), pp. 619–624.
9. Y. Jani, "Optimizing database performance for large-scale enterprise applications," Int. J. Sci. Res. (IJSR) **11**, 1394–1396 (2022).
10. G. Oddi, M. Panfili, A. Pietrabissa, *et al.*, "A resource allocation algorithm of multi-cloud resources based on markov decision process," in *2013 IEEE 5th international conference on cloud computing technology and science,* vol. 1 (IEEE, 2013), pp. 130–135.
11. A. Alsarhan, A. Itradat, A. Y. Al-Dubai, *et al.*, "Adaptive resource allocation and provisioning in multi-service cloud environments," IEEE Trans. on Parallel Distributed Syst. **29**, 31–42 (2017).
12. K. Alhamazani, R. Ranjan, K. Mitra, *et al.*, "Clams: Cross-layer multi-cloud application monitoring-as-a-service framework," in *2014 IEEE International Conference on Services Computing,* (IEEE, 2014), pp. 283–290.
13. D. G. Roy, D. De, M. M. Alam, and S. Chattopadhyay, "Multi-cloud scenario based qos enhancing virtual resource brokering," in *2016 3rd international conference on recent advances in information technology (RAIT),* (IEEE, 2016), pp. 576–581.
14. Y. Jani, "Unlocking concurrent power: Executing 10,000 test cases simultaneously for maximum efficiency," J Artif Intell Mach Learn. & Data Sci 2022 **1**, 843–847 (2022).
15. S. K. Panda and P. K. Jana, "Uncertainty-based qos min–min algorithm for heterogeneous multi-cloud environment," Arab. J. for Sci. Eng. **41**, 3003–3025 (2016).
16. M. Abouelyazid and C. Xiang, "Architectures for ai integration in next-generation cloud infrastructure, development, security, and management," Int. J. Inf. Cybersecur. **3**, 1–19 (2019).
17. S. B. Bhushan and C. P. Reddy, "A qos aware cloud service composition algorithm for geo-distributed multi cloud domain," Int. J. Intell. Eng. Syst. **9**, 147–156 (2016).
18. A. V. Dastjerdi, "Qos-aware and semantic-based service coordination for multi-cloud environments," PhD, Dep. Comput. Inf. Syst. The Univ. Melbourne (2013).