Volume 12, No.8, August 2025

Journal of Global Research in Mathematical Archives

ISSN 2320 – 5822

RESEARCH PAPER

Available online at http://www.jgrma.com

A COMPREHENSIVE SURVEY ON AI-DRIVEN RESOURCE ALLOCATION AND COST OPTIMIZATION TECHNIQUES IN SAAS PLATFORMS

Dr Chintal Kumar Patel[1]

[1] Associate Professor, CSE, Geetanjali Institute of Technical Studies
chintal.patel@gits.ac.in

Abstract: This has been possible through the popularity of Software-as-a-Service (SaaS) platforms in changing the way software is made, offered, and used in a cloud setting. As these platforms ramp up, it is very challenging to manage workloads, satisfy varying user needs and respond to high-service-level agreements (SLAs) on costs and resources. The usual inefficient and inflexible management of resources via the traditional static administration techniques tends to result in underutilization, overprovisioning, and augmented operations costs. This survey gives a review of the Artificial Intelligence (AI)-based approaches to optimizing the allocation of resources and reduction of costs within SaaS ecosystems. It emphasizes the application of machine learning, deep learning, and reinforcement learning to automate workload prediction, intelligent scheduling and online resource scaling. Moreover, it analyzes cost prediction models on AI, SLA-wise optimization plans, and use of cloud-native and third-party tools in assisting financial efficiency. Architectural structures of SaaS platforms, as well as critical resource measurements, including CPU, memory, storage and bandwidth, are discussed in the paper. Current issues such as the interpretability of the models, the complexity of integration, data privacy are discussed, and recommendations regarding the future research directions, including explainable AI, federated learning, and multi-objective optimization, are given. The work intends to assist researchers and practitioners to develop intelligent, adaptive, and cost-effective SaaS systems which strike a balance between performance and economical sustainability.

Keywords: Software-as-a-Service (SaaS), Cloud Computing, Resource Allocation, Cost Optimization, Artificial Intelligence (AI), Machine Learning (ML), Deep Learning, Reinforcement Learning.

1. INTRODUCTION

The emergence of cloud computing has changed the face of software development, deployment and consumption [1]. Software as a Service (SaaS), which enables customers to access software programs over the Internet without the need for complicated infrastructure or local installation, has gained popularity among various service models [2]. Such a model enables the business to scale easily, cut down the initial capital cost and provide ubiquitous access to software services [3]. Digital transformation is in full swing, with SaaS picking up even more speed in such industries as healthcare, finance, education, and manufacturing.

SaaS platforms give the required infrastructure and middleware, which enables software vendors to create, host and manage applications effectively [4]. Such platforms are dynamic environments with varying workloads, varying user demands and demanding service level agreements (SLAs). It is important to have good resource management in these environments in order to maintain quality of services and efficient operations, as well as scale [5]. Nevertheless, the conventional practices in resource distribution involving fixed thresholds and human-based control may not sufficiently implement the real-time needs of SaaS sites, thus causing inefficiencies and performance bottlenecks.

Resource allocation powered by Artificial Intelligence (AI) has become one of the disruptive solutions. Using machine learning (ML), reinforcement learning, and predictive analytics as its methods, AI technology allows platforms to study past data, determine the trends in workload and automatically adjust the amount of computing resources based on those predictions. This is a smart and dynamic method of using resources in the best way possible, minimizing downtimes of the systems and responding to fluctuations in demand, without human control [6]. Due to their ability to learn and improve over time, AI models are especially well-suited to the highly dynamic and shifting context of current SaaS platforms dynamic and adaptive model improves the operational efficiency of highly dynamic operations by reducing the underutilized and over-provisioning of resources [7]. It does not just guarantee optimal utilization of resources but also decreases the downtime of the system, increases the reliability of services, and eases the experience of the user [8]. In addition, AI-based applications are able to automatically adapt to unreliable workload fluctuations and user patterns without manual control or pre-written protocols, saving administrative resources.

The need to optimize costs has emerged as a priority of organizations working with cloud-based environments, alongside effective utilization of resources [9]. With the expansion of SaaS providers, cloud costs may easily blow up, unless there is careful control of those costs. It has been reported that companies tend to spend huge sums of money underutilized resources, overprovisioning and the inability to see cloud consumption patterns [10]. Cost optimization techniques using AI can assist in these issues through solutions like smart workload scheduling, cost-aware provisioning and predictive budgeting [11]. Such techniques do not only improve the financial performance, but also make the businesses sustainable in the ever-cost conscious market. The AI models have the potential to predict the usage patterns and propose the best resource upscaling to prevent unwanted overcommitment. Cost-aware workload scheduling ensures that jobs are executed in the most cost-efficient manner.

1.1. Structure of the Paper

This paper is organized as follows: Section 2 overview of Software as a service (SaaS). Section 3 AI technique for resource allocation in SaaS Section 4. AI-Driven cost optimization strategies in Section 5 Literature review, Section 6 Conclusions and future work.

2. OVERVIEW OF SAAS PLATFORM

A cloud-based software delivery and licensing paradigm known as Software as a Service (SaaS) involves the supplier hosting and enabling end users to access the program using a web interface, and maintaining it on the cloud [12]. Software delivery, deployment refer to the process of making the software available to users, resulting in A client-side application that is ready to use [13]. The procedures that take place between purchasing and executing software can also be referred to as software deployment.

Figure 1: SaaS diagram in Cloud Computing

A business concept known as Software as a Service, or SaaS, involves hosting software applications in the cloud and making them accessible via a range of devices, including networks, PCs, and mobile phones (see Figure 1). It connects core components such as app servers, databases, and code, enabling seamless software delivery and usage over the internet.

2.1. Architecture of SaaS platforms

SaaS architecture must take into account MTA, scalability, quick development, and customization. According to Figure 2, The four main SaaS architectures are PaaS-based, middleware-based, database-oriented, and service-oriented SaaS.

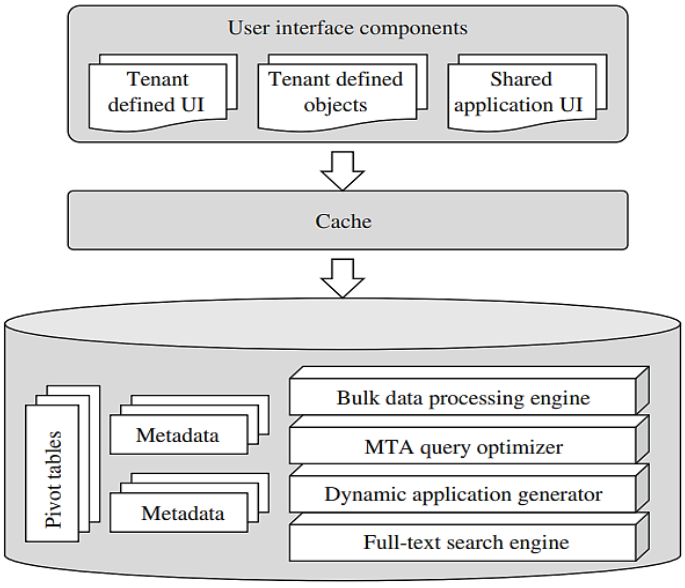

Figure 2: Architecture of SaaS platform

Here is the architecture of SaaS platforms as follows:

2.1.1. Database-oriented SaaS:

Salesforce is a database-oriented architectural system that enables developers to write code and link it to GUI services and other software elements, hence facilitating customization. Every tenant uses the same database and schema as part of the multi-tenant

architecture (MTA) [14]. To ensure scalability, a two-level mechanism is used: A second-level scheduler allocates tasks to stateless servers inside each cluster, while a top-level scheduler uses tenant information to route user requests to various clusters. The database on both servers is the same, and data allocation can be improved via optimization methods.

2.1.2. Middleware-based Approach

The middleware-based approach is shown by Corentech.com, which prioritizes rapid development over deep customization. It enables traditional software to be quickly transformed into multi-tenant aware (MTA) software within hours by modifying only the data access layer, as illustrated in Figure 3. This approach significantly reduces redevelopment time while maintaining compatibility with existing application logic.

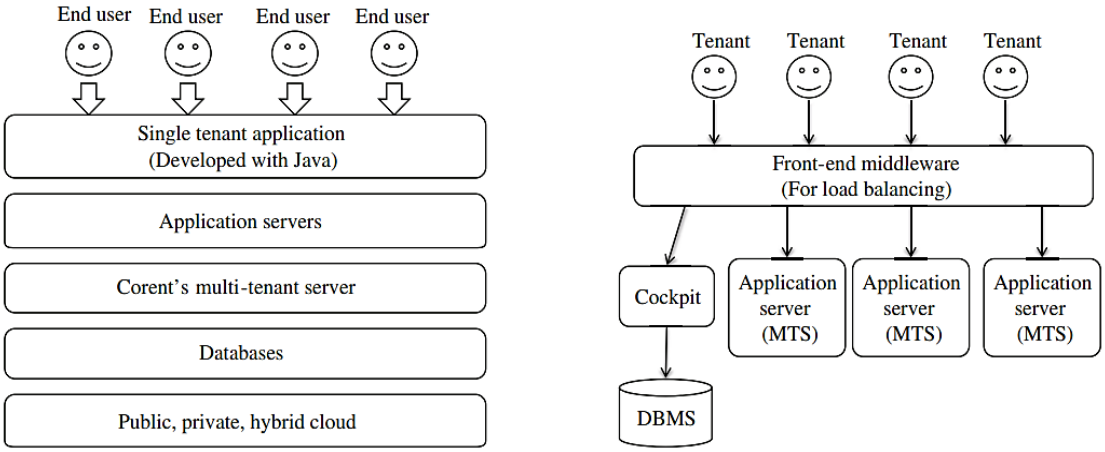

Figure 3: Middleware-based Approach

The architecture utilizes a kernel to manage the MTA layer, redirecting database access requests to the appropriate databases, thereby allowing for efficient migration with minimal changes to existing applications and databases. Scalability is achieved through a two-level mechanism, where the top-level scheduler distributes tenant requests across clusters. This approach preserves existing database properties, such as ACID, and reduces security concerns compared to architectures like Salesforce.com, as tenants do not share databases or schemas.

2.1.3. Service-oriented SaaS:

Easy SaaS is a prominent example of a service-oriented SaaS system that employs Service-Oriented Computing (SOC) principles to design and manage both tenant-specific applications and underlying infrastructure services, as seen in Figure 4. This design makes it possible for services to be flexible, modular, and reusable, allowing easy integration and customization based on tenant needs. It also supports dynamic service discovery and loose coupling, which enhances scalability and maintainability.

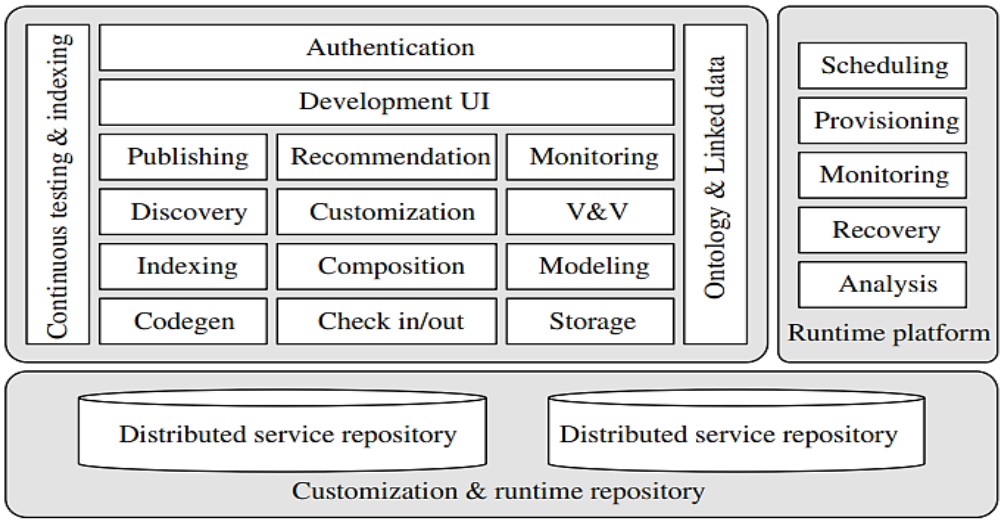

Figure 4: service-oriented SaaS

It supports extensive customization by enabling developers to compose tenant applications using reusable GUI, workflow, service, and data components, or by creating new ones [14]. A consumer-oriented approach, incorporating application templates and a recommendation system (similar to the grapevine approach), enhances component selection [15]. Easy SaaS supports various MTA

designs, including shared or separate databases per tenant, and ensures scalability through a two-level mechanism, combined with features such as automated migration.

2.1.4. PaaS-based Approach:

In order to provide multi-tenant aware (MTA) SaaS applications, the PaaS-based method depends on the underlying Platform as a Service's (PaaS) characteristics. This includes built-in features such as automated scheduling, fault tolerance, scalability, and resource management, as depicted in Figure 5. This approach simplifies development by offloading infrastructure concerns to the PaaS provider, enabling faster deployment and easier maintenance.

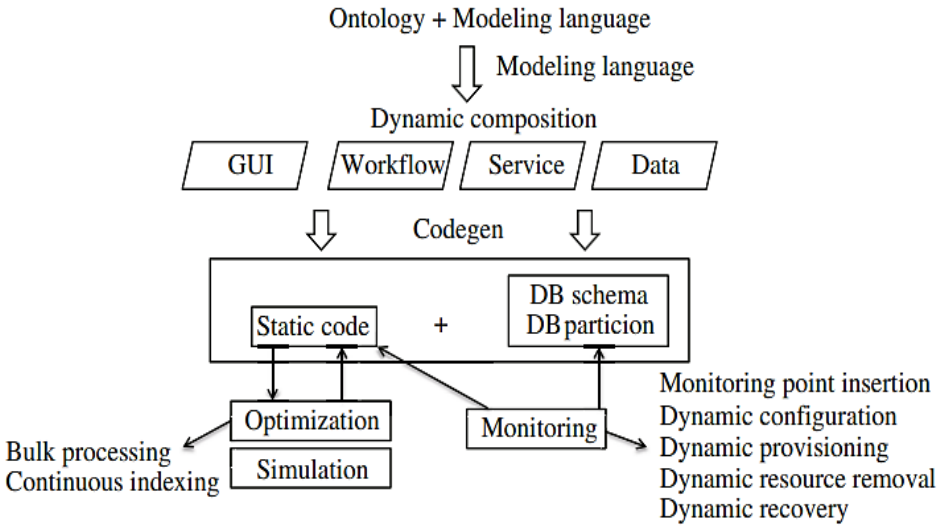

Figure 5: PaaS based approach

The SaaS developer must create the customization feature [16]. Current PaaS frequently prevents direct access to the database or data storage underneath. Every PaaS has its own features for supporting MTA. The scaling mechanism of the underlying PaaS determines SaaS applications' salability when using a PaaS-based approach. GAE supplies the namespace for MTA, and each tenant is individually recognized.

2.2. Cloud Resource Management

The systematic distribution and administration of virtual infrastructure is known as cloud resource planning, and it guarantees the best possible application performance, cost effectiveness, and scalability [17]. The main elements of this planning procedure usually consist of:

2.2.1. CPU

The crucial planning parameter is the quantity of virtual CPUs (CPUs) needed to handle workloads. It impacts application processes' parallelism and performance [18]. While over-provisioning results in needless expenses, under-provisioning causes latency and poor performance..

2.2.2. Memory RAM

Memory planning guarantees that programs have enough RAM to handle data in-memory. Application crashes and dependency on sluggish swap storage might be caused by insufficient memory, whereas idle Data Transmission Methods are the result of excessive memory allocation.

2.2.3. Storage

Choosing between block, object, or file storage is part of storage planning, as is making sure there is enough capacity and IOPS (input/output operations per second). Planners must take backup procedures, retention requirements, and access frequency into account to prevent storage inefficiencies.

2.2.4. Bandwidth

In order to enable traffic to and from cloud services, bandwidth planning entails making sure there is sufficient entry and egress capacity. Application performance and user experience may be impacted by throttling, latency, or high outbound data transfer charges.

2.3. Challenges Related to Cloud in SaaS

These challenges must be addressed to ensure reliable performance and secure delivery of services in cloud environments challenges are discussed below:

- **Security:** The majority of SMEs take use of common cloud space offerings and employ SaaS [19], Security is seen as one of the major obstacles to implementing SaaS cloud computing, which impacts data security and piracy vulnerabilities.
- **Privacy:** In SaaS cloud computing, data privacy describes how cloud services protect private information from potential attackers. It is accomplished by evaluating user behavior in relation to the usage model.
- **Data Confidentiality:** Data confidentiality in cloud computing refers to the use of authentication and access control techniques to safely store private and extremely sensitive information, including bank account details and various access passwords.
- **Quality Assurance Transparency:** Strong agreements between legal requirements are necessary for thorough and transparent audits of ongoing evaluations of various services rendered. After implementing cloud-based security, the majority of major organizations always find it difficult to assess the corporate organization's overall performance.
- **The risks of the SaaS security model:** SaaS-based apps are subject to data security plans. SaaS is carefully crafted with client needs in mind [20]. DDOS attack confirmation cloud organization, botnet area and checking cloud organization, cloud page isolation and anti-contamination application, content security cloud organization, security scene noticing, and recommended cloud organization are frequently cited examples of SaaS [21], cloud spam filtering and neutralizing activity, etc.

3. AI TECHNIQUE FOR RESOURCE ALLOCATION IN SAAS

Software-as-a-Service (SaaS) platforms' computing resources. Effective network distribution has a direct impact on cost, scalability, and performance, storage, making resource allocation a crucial issue. Because SaaS workloads are dynamic and elastic, traditional static allocation techniques often fail to effectively handle them. AI methods, especially those based on ML and optimization algorithms, are being used more and more to automate and improve resource allocation, as shown in Figure 6 technique for resource allocation in SaaS.

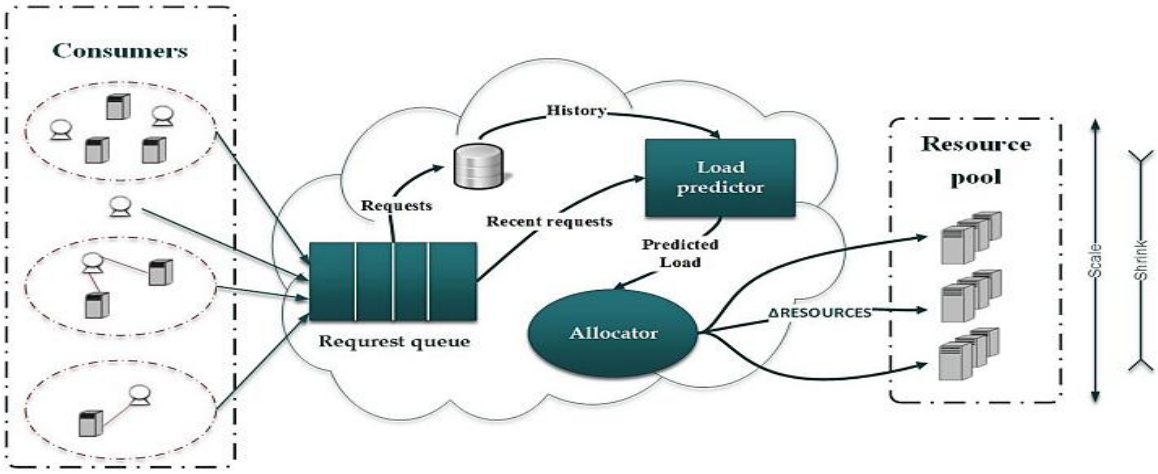

Figure 6: AI technique for resource allocation in SaaS

3.1. Machine Learning Approach

Machine learning enables machines to process and analyze data more efficiently. In many cases, identifying patterns or extracting insights from complex datasets can be challenging for humans. ML solves this by allowing systems to learn from data and gradually enhance their functionality without the need for explicit programming. Extensive research has been conducted to develop algorithms that empower machines to learn autonomously and make intelligent decisions.

3.1.1. Supervised Learning

Algorithms for supervised ML are those that call for outside help a such ML task are model building, model evaluation and tuning, and model deployment into production.

- **Decision Tree:** A decision tree is an organizational chart that displays options and outcomes as a tree. The nodes of the graph represent choices or outcomes.
- **Naïve Bayes:** The NB technique is a straightforward algorithm for probability used in classification that calculates its probability value by combining values and frequencies from the linked collection.

3.1.2. Unsupervised learning

Unsupervised learning learns very little from the data. Using the features, it has already learned, it is able to identify the class of data upon introduction. Its main applications are in feature reduction and clustering. Two main methods for dimensionality reduction and clustering are included:

- **Clustering:** Data is automatically grouped or clustered using this unsupervised learning technique, which is also referred to as grouping or clustering.
- **Principal Component Analysis (PCA):** PCA, reduces the data's dimension. To get a better understanding of PCA Two axes are used to plot the data when it is displayed on a graph. Once PCA is applied, the data reduced to one dimension in Figure 7 before and after PCA.

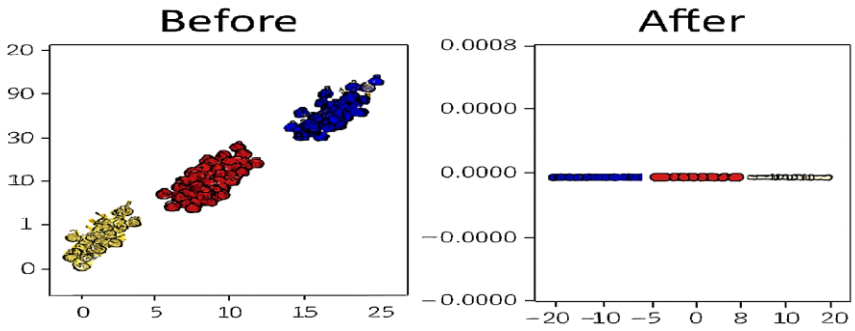

Figure 7: Before and after PCA

3.1.3. Reinforcement Learning

A powerful AI method for handling dynamic and complex situations, such as SaaS systems, is reinforcement learning, or RL. Because of changes in infrastructure, application load, and user behaviour, resource consumption varies in these settings. While traditional rule-based or static techniques are unable to effectively adjust to these changes, reinforcement learning (RL) offers a self-learning framework that enables optimal judgments in real-time through ongoing system interaction. The two approaches in reinforcement learning are below:

- **Q learning:** A model-free method used for learning the value function when system dynamics are unknown.
- **Deep Q-Networks (DQN):** To manage a vast state action space, combine Q learning with deep neural networks, useful in complex SaaS system.

3.2. Deep Learning for Complex Pattern Recognition

SaaS platforms operate in highly dynamic environments with complex usage behavior, making it challenging to model and manage resource demand using traditional methods. DL has emerged as a powerful tool for identifying intricate patterns and long-term relationships in multidimensional data and extensive time series, making it particularly well-suited for predictive resource allocation and cost optimization in SaaS systems.

- **Long Short-Term Memory (LSTM):** The Recurrent Neural Network (RNN) variant known as LSTM was created especially for use with sequence data, including time-series data, text, and voice. RNNs are unable to retain long-term associations because to the vanishing gradient issue.
- **CNN:** CNNs are specially crafted networks to process spatial and image data. They possess convolutional layers, which apply filters to select characteristic features such as shapes, textures, and edges [22]. CNNs compare well with ordinary fully connected networks in that they reduce parameters using shared weights, which is particularly suitable for large-scale image recognition problems.

4. AI-DRIVEN COST OPTIMIZATION STRATEGIES

The rise of SaaS has led to the development of dynamic, scalable systems that require intelligent management of infrastructure costs. Traditional cost optimization strategies often fall short in handling the complex and fluctuating workloads characteristic of SaaS environments [23]. In contrast, AI-driven approaches offer data-driven, adaptive, and automated techniques to optimize costs without compromising performance or SLA (Service Level Agreement) compliance.

4.1. Cost prediction and budget forecasting

Cost prediction and budget forecasting are essential components of cost optimization in SaaS platforms. time-series forecasting, in which patterns are found by examining previous consecutive data [24]. The most popular models for modelling financial time series, such as share prices, market demand, and revenue growth, are the Exponential Smoothing State Space Models (ESSM), Seasonal Autoregressive Integrated Moving Average (SARIMA), and Autoregressive Integrated Moving Average (ARIMA). businesses that prepared their budgets using ARIMA-based models. Because financial data contains intricate, nonlinear interactions, neural networks including deep learning models like Transformer and Long Short-Term Memory (LSTM) networks have attracted a lot of interest in financial forecasting [25]. The LSTM network is very useful for long-term budget planning since it can handle sequential financial data.

4.2. SLA-Aware Resource Cost Optimization

SaaS platforms must uphold SLAs that guarantee performance and availability, making cost optimization more complex. Reinforcement learning and multi-objective optimization algorithms have emerged as effective solutions. These approaches dynamically balance resource costs against SLA requirements by learning optimal scaling policies. SLA penalties are integrated into cost functions, enabling systems to adapt in real-time while minimizing the risk of service degradation or contract violations to manage resource allocation more efficiently. AI-driven systems can be employed to predict workloads and adjust resources in real-time. Dynamic resource allocation techniques [26], such as load balancing and elastic scaling, can ensure that resources are optimally distributed, avoiding underutilization or overloads that could breach SLAs.

4.3. Cost Optimization Tools and Frameworks

Cost optimization in SaaS platforms is increasingly supported by specialized tools and frameworks that leverage artificial intelligence and ML [27]. These tools provide actionable insights, automate decision-making, and integrate with cloud infrastructure to help organizations maintain financial efficiency while ensuring performance and compliance with SLAs, are included:

4.3.1. Cloud-Native Cost Optimization Tools

Exclusive cloud-native cost optimization techniques are offered by major cloud service providers including Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP) [28]. These tools are tightly integrated into the cloud ecosystem and leverage AI and ML to analyze usage data, generate insights, and recommend cost-saving actions in real-time.

4.3.2. AWS Cost Explorer & Compute Optimizer

AWS Cost Explorer and Compute Optimizer leverage historical usage data and machine learning to provide recommendations for optimal instance types, rightsizing opportunities, and cost-effective reserved instance purchases.

- Azure Cost Management and Azure Advisor provide budget tracking, anomaly detection, and performance vs. cost recommendations. Advisor suggests underutilized resources and alternative configurations.
- Google Cloud Recommender: Analyses compute, storage, and network usage to generate cost and performance optimization suggestions using AI algorithms.

4.3.3. Open-Source and Third-Party Frameworks

Cloud-native tools offer built-in cost management within their ecosystems, open-source and third-party frameworks provide greater flexibility, multi-cloud support**,** and advanced analytics features.

- **Kubecost:** An open-source tool specifically designed for Kubernetes environments. It provides real-time cost visibility by allocating costs down to the pod, namespace, service, or label level.
- **Cloud Zero:** A SaaS-based platform that emphasizes cloud cost intelligence. It correlates cost data with engineering activity to offer contextual insights into spend.

5. LITERATURE OF REVIEW

This section presents earlier studies on AI-driven resource allocation and cost optimization techniques in SaaS environments. Table 1 provides a structured comparison of previous research, focusing on AI-driven resource allocation and cost optimization strategies within SaaS platforms and their associated challenges.

Wang and Yang (2025) present the concept of an intelligent resource allocation system that dynamically schedules it using reinforcement learning (DQN) and employs DL (LSTM) to forecast requests. Because the suggested system properly forecasts resource requirements and allows for real-time modifications to these requirements, it boosts resource use by 32.5%, decreases average reaction time by 43.3%, and lowers costs by 26.6%. Experimental findings in a production cloud environment show that the suggested approach is very effective and capable of guaranteeing great service quality. The paper presents a scalable approach to an intelligent management of cloud resources that is effective and can be used as a source of knowledge in future optimization of cloud systems [29].

Kumar et al. (2024) demonstrate that near-zero-time resource allocation is possible with an AI-optimized allocation of resources that satisfies the requirements of a dynamic financial environment, utilizing AI-based decision-making (ADDM) and a Dynamic Nested Neural Network (DNN) in resource allocation. This is achieved in combination with a Markov Decision Process (MDP) that enables probabilistic decisions, maximizing computing power, storage, and network resources in a cloud environment to create an intelligent and adaptive system. This improves efficiency, security, and responsiveness, enabling transactions to be performed in a real-time, seamless, and reliable manner. Its outcomes show faster transaction rates, efficient utilization of resources and a strong security system that are of the essence in financial transactions [30].

Singhal et al. (2024) AI-driven optimization algorithms to enhance overall system performance. Although traditional algorithms like round-robin and heuristic-based methods have been widely studied, they face limitations in scalability, adaptability, and energy efficiency, leaving many challenges unresolved. research proposes a novel AI-driven load balancing framework that integrates machine learning models with optimization algorithms to dynamically adjust workloads in real-time, optimizing resource utilization and reducing latency. This innovative approach represents the first of its kind in applying AI for real-time decision-making in cloud load balancing, providing a more adaptive and efficient solution [31].

Manchana (2024) optimization of cloud costs, encompassing cloud pricing, analysis, and resource allocation tactics. This paper's analysis highlights important cost optimization options across key cloud providers (AWS, Azure, and GCP), which shows that implementing cloud cost optimization strategies may result in considerable cost reductions for organizations using cloud-native SaaS setups. It explores the FinOps framework and how it may be used in real-world manufacturing through a case study. Demonstrating the functions and input of several teams, such as IT, DevOps, Cloud, and Product Engineering [32].

Zhang, Bai and Xu (2023) the best way to distribute cloud resources when demand and supply are unpredictable for SaaS companies. This strategy ensures that the SaaS provider's revenue is maximized while maintaining QoS constraints, depending on the SaaS level. It also makes it easier for the IaaS provider to precisely assign and utilize all of the IaaS resources. The approach not only creates

quantitative demand and resource models, but it also recommends the optimal IaaS resource allocation strategies in three distinct scenarios: uncertain demand and certain supply, certain demand and uncertain supply, and uncertain demand and uncertain supply. Experiments confirm the efficacy and efficiency of the three algorithms, and the findings demonstrate that, in the context of concurrently uncertain supply and demand, The SaaS provider's income and IaaS resource utilisation are effectively increased without violating the QoS constraint [33].

Zheng, Pan and Liu (2021) in order to carry out customers' tasks, SaaS companies pay IaaS providers for on-demand instances. Because IaaS instances are pay-as-you-go, SaaS firms are able to purchase and release instances as needed. SaaS providers must choose if and when to release an idle instance in order to optimize. SaaS companies use the arrivals and execution time cost optimization technique to assist them decide whether to release instances [34].

Wang (2020) to enable basic SaaS services, cloud-based SaaS service platforms must provide features including application virtualization, standardizing data formats, exchanging information, automating deployment and distribution, and managing user and generation operations effectively. To put it simply, the unified service support platform is a cloud computing platform. The cloud-based unified SaaS service platform has to have the following characteristics in order to allow SaaS services: data format standardization, information exchange, application virtualization, automated deployment and delivery, and efficient user and operation management. The unified service support platform is essentially a cloud computing platform [35].

Table 1: Comparative Analysis of AI-Driven Resource Allocation and Cost Optimization Literature in SaaS Platforms

Reference	Focus Area	Key Findings	Challenges	Key Contribution
Wang et al. (2025)	Intelligent resource allocation using LSTM and DQN	increased the use of resources by 32.5%. 43.3% shorter average response time and 26.6% lower operating expenses	Handling dynamic workloads, Real-time scheduling, Forecasting demand accurately	Proposed a hybrid model combining LSTM for demand forecasting and DQN for dynamic scheduling - Validated in a real-world production cloud environment. Demonstrated scalable and efficient resource management approach
Kumar et al. (2024)	AI-driven resource allocation in cloud financial systems	A Markov Decision Process (MDP)-enabled Dynamic Nested Neural Network (DNN) with AI was suggested for security, resource efficiency, transaction speed, and flexibility.	Handling real-time demands in dynamic financial systems; integrating security with efficiency.	Introduced a hybrid DNN-MDP model for adaptive allocation and security enhancement in financial cloud platforms.
Singhal et al. (2024)	AI-based real-time load balancing	Developed an AI-driven load balancing framework integrating machine learning with optimization algorithms, and improving adaptability and energy efficiency.	Scalability, energy efficiency, and adaptability in traditional methods.	First real-time AI-enabled load balancing system for cloud resource optimization.
Manchana et al. (2024)	Cost-cutting techniques for cloud-native SaaS	Emphasized in cloud-native SaaS. Showcased significant cost savings using strategic analysis and optimization across AWS, Azure, and GCP.	Managing cost in multi-provider environments; team coordination.	Provided a real-world case study showing practical FinOps application in a manufacturing SaaS environment.
Zhang et al. (2023)	Allocating resources in an uncertain environment	Proposed strategies for optimal IaaS resource allocation under uncertain supply/demand. Introduced quantitative models ensuring QoS adherence while maximizing SaaS provider revenue.	Resource uncertainty; maintaining QoS under variable conditions.	Developed algorithms for three uncertainty scenarios, improving utilization and revenue while ensuring QoS compliance.
Zheng et al. (2021)	Instance release decision in SaaS	Focused on on-demand IaaS resource usage. Developed cost optimization algorithms to determine	Decision-making under uncertain job arrivals and execution times.	Helped SaaS providers make smarter instance release decisions to reduce costs.
Wang et al. (2020)	SaaS service management via unified platform	Described a unified SaaS support platform enabling app virtualization, data standardization, automated deployment, and effective user management based on cloud computing.	Need for standardized, efficient management infrastructure in SaaS.	Proposed a service platform structure essential for scalable and efficient SaaS delivery.

6. CONCLUSION AND FUTURE WORK

AI-powered methods for Software-as-a-Service (SaaS) platform resource allocation and cost optimization. As cloud-based applications become increasingly dynamic and complex, traditional static approaches to resource management and budgeting are proving insufficient. AI techniques particularly ML, DL, and reinforcement learning offer adaptive, data-driven strategies to automatically allocate resources based on predicted workload demands and optimize operational costs without compromising performance or SLA compliance. These intelligent methods reduce underutilization and overprovisioning, enhance system reliability, and improve the user experience. In addition to discussing AI methodologies, reviewed several cloud-native and third-party tools that support cost optimization through actionable insights and automation. The integration of AI systems into diverse cloud infrastructures can also be complex due to heterogeneous configurations and data privacy concerns.

The development of explainable AI (XAI) methods should be investigated in future research to improve transparency, particularly in decision-making procedures. Furthermore, safe and traceable resource management may be achieved by fusing AI with cutting-edge technologies like blockchain. Research into federated learning could enable decentralized, privacy-preserving optimization across multi-tenant SaaS platforms. Finally, more attention should be directed toward multi-objective optimization frameworks that jointly consider cost, performance, energy efficiency, and security. As SaaS adoption continues to grow, advancing AI capabilities will be pivotal in building intelligent, efficient, and sustainable cloud services.

REFERENCES

[1] S. Garg, "Predictive Analytics and Auto Remediation using Artificial Inteligence and Machine learning in Cloud Computing Operations," *Int. J. Innov. Res. Eng. Multidiscip. Phys. Sci.*, vol. 7, no. 2, 2019.

[2] H. He, "Applications deployment on the SaaS platform," in *5th International Conference on Pervasive Computing and Applications*, IEEE, Dec. 2010, pp. 232–237. doi: 10.1109/ICPCA.2010.5704104.

[3] J. Lee, "A view of cloud computing," *Int. J. Networked Distrib. Comput.*, 2013, doi: 10.2991/ijndc.2013.1.1.2.

[4] K. Kim and K. Lee, "An Implementation of Open Source-Based Software as a Service (SaaS) to Produce TOA and TOC Reflectance of High-Resolution KOMPSAT-3/3A Satellite Image," *Remote Sens.*, vol. 13, no. 22, 2021, doi: 10.3390/rs13224550.

[5] S. Garg, "AI/ML Driven Proactive Performance Monitoring, Resource Allocation and Effective Cost Management in SAAS Operations," *Int. J. Core Eng. Manag.*, vol. 6, no. 6, pp. 263–273, 2019.

[6] U. M. R. Inkollu and J. K. R. Sastry, "AI-driven reinforced optimal cloud resource allocation (ROCRA) for high-speed satellite imagery data processing," *Earth Sci. Informatics*, 2024, doi: 10.1007/s12145-024-01242-5.

[7] V. Shah, "Managing Security and Privacy in Cloud Frameworks: A Risk with Compliance Perspective for Enterprises," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 06, pp. 1–13, 2022, doi: https://doi.org/10.14741/ijcet/v.12.6.16.

[8] A. S. Elgamal, O. Z. Aletri, B. A. Yosuf, A. Adnan Qidan, T. El-Gorashi, and J. M. H. Elmirghani, "AI-Driven Resource Allocation in Optical Wireless Communication Systems," in *International Conference on Transparent Optical Networks*, 2023. doi: 10.1109/ICTON59386.2023.10207473.

[9] S. Deochake, "Cloud Cost Optimization: A Comprehensive Review of Strategies and Case Studies," *SSRN Electron. J.*, 2023, doi: 10.2139/ssrn.4519171.

[10] K. Murugandi and R. Seetharaman, "A Study of Supplier Relationship Management in Global Procurement : Balancing Cost Efficiency and Ethical Sourcing Practices," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 2, no. 1, pp. 724–733, 2022, doi: 10.48175/IJARSCT-7744B.

[11] R. Mohite, R. Kanthe, K. S. Kale, D. N. Bhavsar, D. N. Murthy, and R. A. D. Murthy, "Integrating Artificial Intelligence into Project Management for Efficient Resource Allocation," *Int. J. Intell. Syst. Appl. Eng.*, 2024.

[12] I. Ashraf, "An Overview of Service Models of Cloud Computing," *Int. J. Multidiscip. Curr. Res.*, vol. 2, no. 2014, 2014.

[13] V. Shah, "Securing the Cloud of Things : A Comprehensive Analytics of Architecture , Use Cases , and Privacy Risks," vol. 3, no. 4, pp. 158–165, 2023, doi: 10.56472/25832646/JETA-V3I8P118.

[14] W. T. Tsai, X. Y. Bai, and Y. Huang, "Software-as-a-service (SaaS): Perspectives and challenges," *Sci. China Inf. Sci.*, 2014, doi: 10.1007/s11432-013-5050-z.

[15] A. Khanjani, W. N. Wan, and A. A. Abd Ghani, "Feature-based analysis into the trend of software technologies from traditional to Service Oriented Architecture and SAAS cloud," *J. Comput. Sci.*, 2014, doi: 10.3844/jcssp.2014.2408.2414.

[16] H. Guo, J. Li, Z. Du, and M. Li, "PAAS: A protocol-based approach to adaptive service composition," in *ICCASM 2010 - 2010 International Conference on Computer Application and System Modeling, Proceedings*, 2010. doi: 10.1109/ICCASM.2010.5619430.

[17] G. Maddali, "An Efficient Bio-Inspired Optimization Framework for Scalable Task Scheduling in Cloud Computing Environments," *Int. J. Curr. Eng. Technol.*, vol. 15, no. 3, pp. 229–238, 2025.

[18] V. N. Tsakalidou, P. Mitsou, and G. A. Papakostas, "Machine Learning for Cloud Resources Management—An Overview," in *Lecture Notes on Data Engineering and Communications Technologies*, 2023. doi: 10.1007/978-981-19-3035-5_67.

[19] A. Mamdouh, A. Ibrahim, N. S. Abdullah, and M. Bahari, "Software as a Service Challenges : A Systematic Literature Review Software as a Service Challenges : A Systematic Literature Review," no. November 2024, 2022, doi: 10.1007/978-

3-031-18344-7.

[20] B. Sridevi, “Review on challenges in saas model in cloud computing,” *J. Innov. Dev. Pharm. Tech. Sci.*, vol. 3, no. 3, pp. 1–4, 2019.

[21] D. Patel, “Zero Trust and DevSecOps in Cloud-Native Environments with Security Frameworks and Best Practices,” *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 3, 2023.

[22] V.Mahalakshmi and V. Poornima, “Cloud Resource Allocation using Deep Learning Techniques –A Study,” in *2024 4th International Conference on Soft Computing for Security Applications (ICSCSA)*, 2024, pp. 455–459. doi: 10.1109/ICSCSA64454.2024.00079.

[23] Abhishek and P. Khare, “Cloud Security Challenges: Implementing Best Practices for Secure SaaS Application Development,” *Int. J. Curr. Eng. Technol.*, vol. 11, no. 06, pp. 669–676, Nov. 2021, doi: 10.14741/ijcet/v.11.6.11.

[24] R. Changalva, “Predictive Budgeting and Planning with AI in Oracle EPM : Automating Financial Projections,” pp. 4022–4040, 2024.

[25] U. Khairani and N. Rahmani, “Operational Cost Budget Optimization Analysis to Improve Efficiency of PT Citra Robin Sarana,” *Quant. Econ. Manag. Stud.*, vol. 5, pp. 1005–1010, 2024, doi: 10.35877/454RI.qems2813.

[26] R. Williams, “Challenges in Achieving SLA-Aware Operational Efficiency in AI-Driven Service Chains,” no. August, 2024.

[27] R. P. Sola, N. Malali, and P. Madugula, *Cloud Database Security: Integrating Deep Learning and Machine Learning for Threat Detection and Prevention*. Notion Press, 2025.

[28] S. S. S. Neeli, “Serverless Databases : A Cost-Effective and Scalable Solution,” *IJIRMPS*, vol. 7, no. 6, 2019.

[29] Y. Wang and X. Yang, “Intelligent Resource Allocation Optimization for Cloud Computing via Machine Learning,” *Adv. Comput. Signals Syst.*, vol. 9, no. 1, pp. 55–63, 2025, doi: 10.23977/acss.2025.090109.

[30] M. L. Kumar, M. Amanullah, R. Krishnamurthy, N. M. Suganthi, and S. A. Kalaiselvan, “Revolutionizing Financial Cloud Services: AI and Blockchain-driven Resource Allocation for Maximized Transaction Speed and Security,” in *2024 Third International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, 2024, pp. 1–5. doi: 10.1109/ICEEICT61591.2024.10718396.

[31] A. Singhal, P. K. Goel, D. Garg, and C. Sharma, “Enhancing Cloud Performance with AI-Driven Load Balancing and Optimization Algorithms,” in *2024 4th International Conference on Advancement in Electronics & Communication Engineering (AECE)*, IEEE, Nov. 2024, pp. 1254–1259. doi: 10.1109/AECE62803.2024.10911072.

[32] R. Manchana, “Driving Cloud Cost Efficiency: A Collaborative FinOps Approach for Cloud-Native SaaS,” *J. Artif. Intell. Cloud Comput.*, vol. 3, no. 1, pp. 1–8, Feb. 2024, doi: 10.47363/JAICC/2024(3)E129.

[33] L. Zhang, J. Bai, and J. Xu, “Optimal Allocation Strategy of Cloud Resources With Uncertain Supply and Demand for SaaS Providers,” *IEEE Access*, vol. 11, pp. 80997–81010, 2023, doi: 10.1109/ACCESS.2023.3300735.

[34] B. Zheng, L. Pan, and S. Liu, “An Online Cost Optimization Algorithm for IaaS Instance Releasing in Cloud Environments,” in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, 2021, pp. 463–469. doi: 10.1109/CCWC51732.2021.9375937.

[35] W. Wang, “Data Security of SaaS Platform based on Blockchain and Decentralized Technology,” in *2020 International Conference on Inventive Computation Technologies (ICICT)*, 2020, pp. 848–851. doi: 10.1109/ICICT48043.2020.9112421.

