IJMSERH

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

Volume 12, Issue 1, January-March 2024

Impact Factor: 9.274

ijmserh@gmail.com

www.ijmserh.com

International Journal of Multidisciplinary and Scientific Emerging ResearcH (IJMSERH)

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

Architecting Autonomous Cloud Platforms with AI-Driven Self-Optimization Capabilities

Venkatramana Reddy Panyala

Production Engineer, Yahoo, United States of America

ABSTRACT: It is with the introduction of sophisticated cloud computing systems that sophisticated operating environments are being built wherein intelligent automation is a must. This paper develops the architectural design of developing smart cloud platforms capable of self-optimizing using artificial intelligence methods. It is proposed that the suggested framework deployed machine learning, real-time telemetry systems, and decision engines with constraints to create smart cloud platforms capable of analyzing usage trends and re-configuring resources on-the-fly. It has a five-layer architecture, which is infrastructure layer, monitoring layer, decision making layer, actuation layer and learning layer. Theoretical study reveals that the framework has the following advantages: it reduces overheads, optimizes the use of resources, improves services, and minimizes the cost of infrastructure. The architecture is cloud native, having its foundation in reinforcement learning and other machine learning and orchestration concepts.

KEYWORDS: Cloud computing, Autonomous Systems, AI-driven optimization, machine learning, Self-healing infrastructure, Kubernetes, microservices, reinforcement learning, resource management, cloud orchestration.

I. INTRODUCTION

Today's cloud-based architectures have evolved into very complex and distributed multi-tenant infrastructures capable of supporting millions of concurrent users in geographically spread out data centers. As more and more complexities arise in managing these highly scalable architectures, the traditional methods of managing an architecture have become ineffective. This is an inability to respond effectively to workload changes, hardware component failures, security breaches and even configuration changes in time in the case of infrastructure engineers. This causes inefficiencies, expensive and low quality of services.[1]

Machine learning and AI are a game changer in these issues. Essentially, the capability to instill intelligent decision-making capabilities in cloud management infrastructures will allow operators to offload the burden of making routine decisions to intelligent agents that can make correct predictions using previous data and act appropriately automatically. The shift in the cloud operations towards reactive to proactive, through the application of AI, is one of the most radical changes in the architecture of cloud computing.

Prior works on autonomic computing have mentioned the concept of self-optimizing infrastructures [2]. This field of research was born in IBM in the first ten years of 2000s, and it exhibited a new vision of systems that could administer themselves according to high-level policies that were made by themselves by their administrators. Regrettably, the realization of such a vision has been slowed down by the inability to compute and the absence of data. The existence of deep learning and high-throughput stream processing, coupled with cloud-native orchestration frameworks have lastly enabled autonomous cloud management.

We contribute to this problem space in this work in several ways. We present a stratified methodology of cloud autonomous platform design and specify the functionality required of each of the subsystems and how the subsystems should interact with each other. We also elaborate on different AI elements deployed by every tier of the architecture such as the model to be employed, their inputs, and outputs. Moreover, we take into account various engineering issues, which are put into the construction of such systems, and we mainly concern balancing between autonomy and safety. Lastly, we demonstrate how our architecture can be incorporated into cloud-native toolchains, such as Kubernetes, Prometheus, and service meshes.[3]

The remainder of the paper has the following structure. The literature review of autonomous computing, cloud resource management, and machine learning applications will be shown in Section 2. Section 3 will give an overview of the architecture proposal. Section 4 will discuss in more detail the AI and machine learning components of the proposal. Section 5 will discuss some issues and challenges of implementation. Section 6 will discuss the way the future of research is.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

II. RELATED WORK

Guneet Kaur Walia et al., paper includes the thorough analysis of AI-based resource management in fog and edge computing of IoT applications. It emphasizes the benefits of smart scheduling and optimization to enhance latency, energy efficiency, and scalability. Other important issues in research that are addressed by the authors include heterogeneity, security and real time decision making. Future directions focus on adaptive and autonomous resource allocation systems.[4]

Sijing Duan et al., work investigates distributed artificial intelligence that is facilitated by end-edge-cloud cooperation. It details the efficient distribution of computation throughout the layers in order to minimize the latency and bandwidth consumption. The survey finds coordination, data privacy, and model consistency problems. It also describes the prospects of scalable and intelligent distributed systems.[5]

Mohit Kumar et al., paper suggests an autonomic edge-cloud model to predict heart diseases with the help of RF-LRG algorithm. It combines edge computing and machine learning to allow quicker and more precise healthcare analytics. The framework improves the predictability and minimizes the time of response. It demonstrates the effectiveness of edge-assisted AI in medical applications.[6]

Victor Casamayor Pujol et al., represented in the distributed computing continuum, the paper talks of edge intelligence. It points out the new research potentials in applying AI to cloud, edge, and IoT environments. Major areas of focus are real-time analytics, resource orchestration, and system scalability. The paper highlights the importance of effective communication among distributed elements.[7]

Sukhpal Singh Gill et al., this paper discusses how quantum computing can fundamentally transform research and technology. It talks of the ability of quantum algorithms to solve problems that are complex and could not be solved by classical algorithms. The paper mentions the use in optimization, cryptography, and data analysis. It also deals with the existing limitations and future research directions.
[8]

Jolly I. Ogbole1 et al., this paper presents an autonomous cloud infrastructure management generated AI framework. It concentrates on self-learning systems that maximize performance, allocation of resources and fault tolerance. The framework facilitates making decisions proactively with the help of AI models. It emphasizes the possibilities of generative AI in realizing completely automated cloud operations.[9]

Shetty et al. (2024) describe an in-depth method of AI-based autonomous cloud operations based on AIOps. The authors highlight the combination of machine learning models with cloud monitoring systems to allow detecting anomalies proactively, analyzing the root cause, and automating remediation. Their research brings out the role of the

AIOps in mitigating human intervention and enhancing reliability of the systems through continuous learning of operational data. Other issues that affect the effectiveness of autonomous systems, discussed in the paper, include data heterogeneity and model interpretability.[10]

Reddy et al. (2024) offer an in-depth survey of autonomous cloud systems, particularly on self-managing infrastructure. The authors divide different techniques into self-configuration, self-healing, self-optimization, and self-protection. Their work also assesses the current frameworks and establishes some of the major gaps in research such as the necessity to have standardized architectures and better decision-making models. Scalability and adaptability are also highlighted in the survey as the key characteristics of the next-generation cloud platforms.[111]

Chen et al. (2024) investigate AI-based orchestrating resources in cloud-native. Their study illustrates the use of artificial intelligence methods to optimize resource allocation over distributed microservices systems. The authors suggest smart scheduling algorithms which dynamically respond to workload patterns by reallocating resources, thus enhancing performance and minimizing costs. They find that their results are greatly contributing to efficient systems in comparison to the traditional rule-based orchestration techniques.[12]

Gupta et al. (2024) concentrate on dynamic scaling of machine learning to microservices deployed in Kubernetes. The paper presents predictive scaling models that utilize historical workload data to predict changes in demand. With the combination of these models, Kubernetes autoscaling mechanisms, the system will have a higher level of resource

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

utilization and minimized latency. Other issues that are dealt with by the authors are the accuracy of the models and adaptability in real time in highly dynamic environments.[13]

Hassan and Rahman (2024) address the concept of intelligent cloud operations with the help of AIOps, outlining the major techniques and issues. They work in fields like anomaly detection, event correlation and automated incident management. The authors note that although AIOps can greatly improve the efficiency of operations, such aspects as the quality of data, the complexity of integration, and trust in the automated decision-making process are very important. They propose mixed methods that involve integrating human knowledge with AI-powered systems.[14]

Das et al. (2024) discuss the current developments in the field of deep learning to manage autonomously cloud resources. The paper discusses the different architectures of deep learning in workload prediction, anomaly detection, and resource optimization. The authors emphasize the advantage of deep learning models in processing information of high-dimensional and complicated clouds. Nevertheless, they also identify constraints like high computational complexity, and large training data, which may limit real-world application.[15]

III. PROPOSED ARCHITECTURE

3.1 Architectural Overview

The proposed architectural design to construct the intelligent self-driving cloud platform is organized in a multilayer stack of functional subsystems with each layer specializing in the execution of various tasks in the process of managing the platform. Moreover, the architecture uses the separation-of-concerns approach that enables the single subsystems to be enhanced independently and ensures that the subsystems interact correctly with each other and that the overall system behaves consistently.

Figure 1 shows the suggested architecture of the platform. The architecture has a total of six primary layers, namely Cloud Infrastructure Layer, AI Monitoring and Telemetry Engine, Self-Optimization Decision Engine, Policy and Constraint Manager, Autonomous Actuation Layer and Feedback and Learning Loop. The successive layers depend on the lower layers to be functional and offer functionality to upper layers.

The multilayered architecture is deliberately modular in nature. With this model adopted by organizations, AI-driven optimization may be adopted in stages, beginning with the telemetry layer and slowly proceeding to autonomy. This kind of approach will ensure that no unintended actions take place as the system will no longer be manually operated, but rather autonomous in nature.

3.2 Cloud Infrastructure Layer

The Cloud Infrastructure Layer includes the infrastructure resources which the platform provides support to. They are computing resources like virtual machines, containers, and serverless computing resources; storage resources like block storage, object storage, and file storage; network resources like load balancers, firewalls, service meshes and CDN nodes; and platform services like databases, messaging services, and caching

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

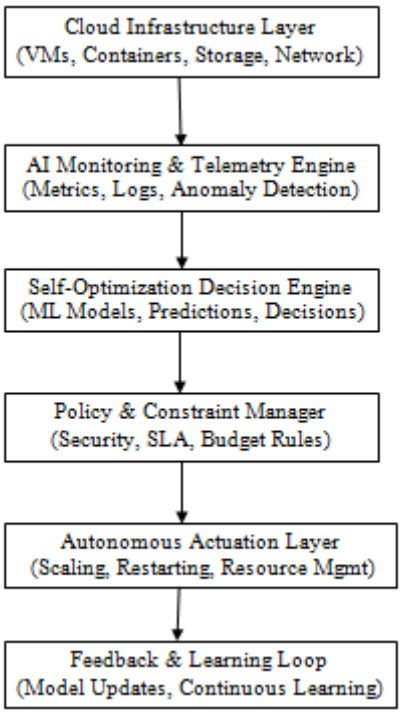

Figure 1: AI-Driven Cloud Platform Architecture

.

The layer offers an API infrastructure management to deal with the heterogeneity and disparity of these kinds of resources and underlying cloud infrastructure providers. At this level, standardization is critical to ensure that the upper levels are cloud agnostic in executing optimization components. The API leverages the Kubernetes' resource APIs extended to cover non-container resources as well as external cloud provider resources [10].

3.3 AI Monitoring and Telemetry Engine.

The AI Monitoring and Telemetry Engine has the role of ensuring that the information of all components in the infrastructure layer is continuously collected, aggregated, enriched and analyzed. Information is received through different channels through high throughput streaming system and processed in real time to offer information that will guide the optimization layer.

This engine makes use of a multi-layered data processing architecture. The lightweight agents are deployed on all the compute nodes in the first layer (edge layer) to collect information locally and conduct early anomaly detection. This reduces the amount of data that needs to be processed centrally. The high-dimensional telemetry (which may be in millions of events per second) is analyzed with stream processing technologies such as Kafka and Flink in the centralized layer.

The telemetry engine contains a time-series database which stores information about operational measurements of different timeframes. The short-term buffer is less than one-second resolution whereas the medium-term buffer is at a resolution of one minute. The long-term buffer is at an hourly resolution. The optimisation of the performance of the system involves each of the buffers.

3.4 Self-Optimization Decision Engine

Self-optimization Decision Engine is the thinking core of the platform. It receives inputs of situation awareness of the telemetry engine and generates optimization plans that can be implemented by the actuation layer. This system is a combination of several dedicated AI engines that carry out different optimizations in various areas, the results of which are subsequently synthesized into a plan of action that can be followed.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

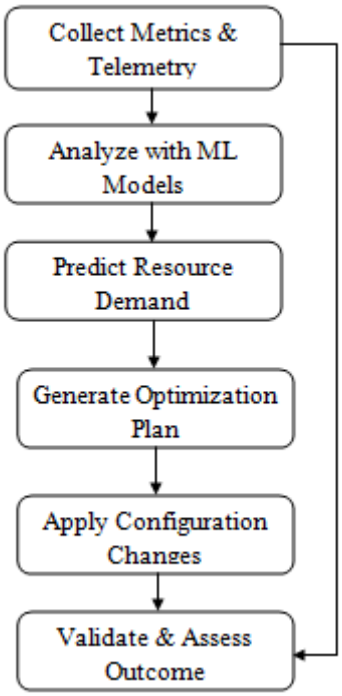

Figure 2: Self-Optimization Decision Loop

The decision engine uses a layered approach to its optimization strategy. The operational layer consists of models that react in less than one second to events such as traffic bursts and hardware breakdowns. Tactical predictive models anticipate alterations of patterns of workload and respond by redistributing resources in minutes and hours. Models in the strategic layer do long-range planning of many aggregated optimization goals, such as reducing total cost of ownership and carbon footprint.

Figure 2 illustrates the optimization feedback loop, which is closed, that drives the Decision Engine. At the first stage, measurements and telemetry are collected in all regions of the infrastructure. The second step will be an analysis of the machine learning algorithms that identify the current condition of the system and any deviation of the ideal performance of the system. Based on previous behavior and seasonal characteristics, an initial solution towards optimization is generated using prediction capabilities of the engine. The policy checks are executed prior to the optimization action being performed and information on the output is the foundation of additional learning.

3.5 Policy and Constraint Manager.
Strict confines have to be adhered to in making decisions in an autonomous manner within a production cloud. The policy and constraint manager is a governance layer in our structure and he/she is in charge of determining the rules and constraints that regulate the actions of the optimization layer. It allows independent activity, which is compliant with policy, service level agreement, regulatory compliance, and organizational risk tolerance levels.

Policy expression language enables administrators to say what is desired of the behavior of the system without necessarily knowing the AI technology employed. There are resource allocation policies that state the resource limits both in terms of minimums and maximums of each workload; scaling policies that restrict the way scaling should be done; cost policies that state the budget limits of resources allocation; availability policies that say how much downtime can be allowed and that infrastructure redundancy requirements; and security policies that accommodate limited access of some portions of the infrastructure.

3.6 Autonomous Actuation Layer
The Autonomous Actuation Layer translates the optimization choices of the decision engine into real changes in the cloud infrastructure. This layer communicates directly with the cloud infrastructure layer, via the unified management API, by making requests to provisioning/de-provisioning of resources, changing parameters, rerouting network traffic, triggering failovers, and any other action required to complete the optimization decision.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

Among the design requirements of the actuation layer is the fact that all autonomous operations that change the production infrastructure have to be reversible and are performed in a manner that has minimal risks. Therefore, the actuation layer must have the ability to detect when an autonomous action starts to have undesired results and roll back the changes made by that action. In order to fulfill this design requirement, the actuation layer follows a two-step execution model wherein changes are initially implemented into a staging environment and tested against expectation before being implemented into the production environment. The actuation layer will employ progressive deployment techniques like canary releases and blue-green deployments in the case where not all autonomous actions can be tested in the staging environment to address any possible risks.

IV. AI AND MACHINE LEARNING ELEMENTS

4.1 Overview of AI Components:

The intelligence of the autonomous system is divided among different AI modules, each responsible for solving a particular optimization problem. The AI modules work as a team, sharing information through the shared knowledge base that is offered by the decision engine. This interaction of the different AI modules and the decision engine is shown in Fig. 3 below.

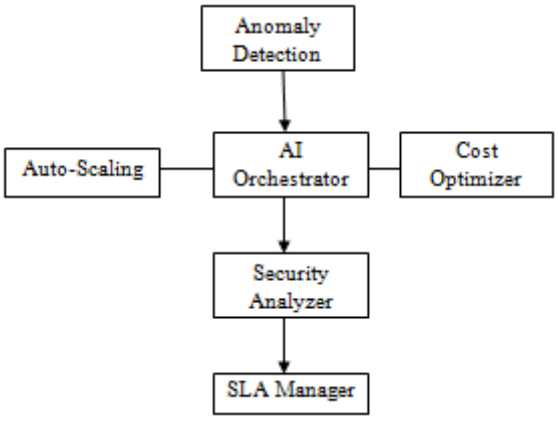

Figure 3:AI Component Interaction Diagram

4.2 Workload Forecasting Models

Precise prediction of future demand of workload is a key to successful proactive optimization of resources. The system consists of a combination of forecasting algorithms that can deal with various types of workload demands in the cloud environment. Such algorithms are the Seasonal Decomposition model, where it is possible to model cyclic behavior of the workload, e.g., daily and weekly change in workload; the Transformer-based sequence model, where it is possible to analyse the non-linear temporal properties of metric data; and the external signals model, where it is possible to consider other variables in the forecast, e.g., marketing campaigns, holiday schedules, and geo-event

The prediction programs are constantly trained with a sliding window of telemetry data. An active process of periodically retraining models is conducted to take into account the potential changes in the behavior of workloads within a certain timeframe. To further support this, online learning algorithm is also given so that it can quickly respond to the sudden shift in the workload behavior that are not identified by the models previously generated. Forecasting algorithms provide probabilistic forecasts by providing deterministic forecasts as well as probability intervals which are subsequently used by the decision making module to measure risk.

4.3 Anomaly Detection Engine

Early detection and prompt reporting of anomalies is important to maintain the health and safety of platforms. The anomaly detection algorithm is based on statistical and machine learning techniques to identify the differences with the usual operational state by examining hundreds of metrics being monitored simultaneously. Multivariate anomaly detection is an Isolation Forests technique [10]. LSTM autoencoders identify temporal anomalies of time-series of individual metrics [4]. Graph neural networks can be used to identify anomalies in the graph of interactions of distributed services.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

False positives reduction is one of the greatest barriers to introducing anomaly detection in cloud system. Excessively sensitive in detection will lead to overload of alerts to the operations teams and mistrust of the autonomous system. To solve this issue, the architecture uses an anomaly filtering pipeline, consisting of several stages of successively more complex processing of anomalies detected. The system re-calculates the probabilistic score of anomaly at each level depending on the new evidence and only autonomous actions are triggered when the score exceeds the set threshold value.

4.4 Reinforcement Learning to Optimize Resources.
The resource allocation in a multi-tenant cloud environment is a challenging combinatorial optimization problem that is defined by a large action space, delayed feedback, and time-dependent environment. Such tasks are particularly well suited to reinforcement learning, which can learn optimal control policies by trial and error, and does not rely on a system dynamics model.

The system makes use of a PPO algorithm-based [11] agent that makes continuous allocation decisions, augmented with model-based planning to utilize workload forecasting models to make rollout predictions about actions' outcomes. Resource utilization measures, the amount of time the pending workload queue has been waiting, the amount of SLA violation currently underway, and the costs incurred in the last time period are some of the features that the agent uses in its decision-making process. The reward function is made up of a mix of efficiency of resource use, SLA compliance rate, and performance/cost ratio, which is weighted based on organizational objectives.

4.5 Cost Intelligence and Finops Integration.
The costs involved in the cloud infrastructure represent a large and unmanaged overhead of large companies. The cost intelligence element of the autonomous system continuously evaluates the trends in resource consumption, identifies ways to save money, and applies them within the policy layer-imposed constraints.[12]

To obtain the detailed resource-level cost information, the component needs to be integrated with cloud provider billing systems, which will enable allocation of costs to individual workloads but to teams as well. The machine learning algorithm will identify underutilized resources that can be either downsized or terminated and potential candidates to be migrated to lower cost instances or regions. [13]It also identifies instances where specific workloads can be good candidates to purchase reserved instances. Cost-saving actions are constantly suggested and prioritized according to anticipated savings, where highly probable, risk-free suggestions are implemented automatically, while bigger actions are manually approved.

V. IMPLEMENTATION-CONSIDERATIONS AND CHALLENGES.

5.1 Data Quality and Observability.
The quality of the data the model is being trained on and takes at the moment is one of the principles of AI optimization, but a lot of other things depend on it. Practically, issues with cloud telemetry data include missing data due to agent failures, clock skew due to variability between distributed data sources, noisy labels to train the model based on historical incidents, and distribution shifts as the workload nature evolves with time.[14]

To address these issues, this architecture instantiates a data quality pipeline in the telemetry engine. The pipeline applies anomaly-aware imputation to fill gaps in the data whilst preventing anomalies that could distort any subsequent data analysis that the model may conduct. The data is synchronized with respect to time series: any clock skews are corrected by synchronizing data based on network time protocols and post facto correlation. Each metric stream is given a data quality score and can be subsequently propagated to the model so that it can make adjustments to its own confidence estimates. The operators can see the data quality scores through an observability dashboard.

5.2. Safety and Human Oversight
The safety considerations associated with the implementation of autonomous systems in production cloud environments are quite different compared with the considerations associated with the safety of the infrastructure systems that are conventionally controlled. Misunderstanding of optimization by the autonomous system might result in numerous issues, such as loss of performance, security breach, or loss of money. The architecture is designed to have many safety measures, which take such fears into account without sacrificing the benefits of autonomy.[15][16]

When it comes to modeling, predictions have uncertainty estimates which affect the confidence levels of the decision engine regarding what to do. High uncertainties in predictions are not considered at the time, but instead they are left

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

unattended till further data is available or sent to the human experts to consider. Policy wise, limits are set on the rate of resource allocation, scale speed and frequency of configuration updates to ensure that optimization runs do not get into self propelled loops. The two-stage actuation process described in Section 3.6 serves as the last defense mechanism in terms of execution.[17]

The human oversight principle will be held by an approach that incorporates hierarchy based escalation, in which all decisions will be escalated to the relevant human authority based on the risk nature involved. Decisions that are low risk and can be confidently made will be implemented automatically without prior notice. Medium risk decisions will result in alerts to be stored to be audited later. Until human authority approves high-risk decisions, these will not be implemented.[18]

5.3 Scalability and Performance Requirements.
The monitoring module and decision-making module of the autonomous system should both be made scalable because the amount of telemetry data collected and optimization decisions required grow in direct correlation with the size of the monitored environment. The platform to handle thousands of services in multiple cloud zones must be able to process millions of metric values per second and guarantee a response time of under a second to optimizations that need a rapid response.

This scalability is made possible through horizontal scaling of stateless modules, implementing partitioned data processing pipelines, which allow telemetry processing in parallel with multiple worker machines, as well as a multi-tier decision-making architecture, where resource-intensive strategies can be computed by background jobs, and fast operations can be executed via high-performance, low-latency communication channels. The platform uses a consistent distributed database to store mission-critical data and policy enforcement data, and eventually consistent databases to store other analysis information.

5.4 Multi-Cloud and Hybrid Environments.
Enterprise organizations typically have a footprint spread over several cloud vendors and also own some infrastructure on-premises. Such diversity should be manageable on the platform and provide users with a similar experience of operation. This level of abstraction in the architecture safeguards the optimization modules against being tainted by vendor specific factors, but in practice, such an abstraction can only be realized by designing carefully the integration points with the providers.[19]

Some of the issues related to autonomous optimization in a multi-cloud environment are the standardization of telemetry information from cloud providers with diverse nomenclature schemes and data gathering periods, the mapping of the resources offered by the providers to a common resource model, enforcing uniformity in policies among cloud providers, and optimization of workload placements in a multi-provider setting based on varying factors such as cost, performance, and compliance.[20]

VI. FUTURE RESEARCH

Although the framework introduced in this paper is an important step towards autonomous cloud management, there are still many open research questions. The following are some of the directions that we find especially significant in future investigation.

Future studies on autonomous cloud optimization aim at bettering privacy, reliability, and trust. Federated learning allows two or more organizations to jointly train models without allowing access to raw telemetry data, promoting greater generalization and maintaining privacy, but issues like heterogeneous workloads and adversarial participants are still. Causal inference is a more powerful method to overcoming the drawbacks of correlation-based machine learning because it establishes the cause-effect relationships, enabling systems to adjust to the changing conditions, even in case of such limitations as a complex causal graph and confounding variables. Moreover, to enhance operator trust and regulatory compliance, it is necessary to increase explainability and interpretability; attention-based models, counterfactual explanations, and interactive query systems are some methods that can make AI decisions more transparent and comprehensible. All these directions are geared towards creating more credible, secure and trustful autonomous cloud ground.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

VII. CONCLUSION

This paper introduces a self-optimizing architecture of autonomous cloud platforms powered by AI. The system is designed in layers: Infrastructure Abstraction, Telemetry Engine, Decision Engine, Policy Manager, Actuation Layer, and an Adaptive Learning Loop. Such a gradual implementation approach allows the gradual adoption and provides efficient functionality in multi-cloud environments.The architecture combines state-of-the-art intelligent cloud computing technologies that include transformer-based forecasting, ensemble anomaly detection, resource optimization through reinforcement learning, and cost intelligence through machine learning. These methods allow anticipating problems, adapting to uncertainties, and ongoing improvement of performance.

The main issues such as data quality, ensuring safety, scalability, and compatibility with multiple clouds are design challenges and not implementation problems. These aspects are critical when it comes to creating credible and trustworthy autonomous cloud systems.These platforms will also be improved in the future. Federated learning will facilitate shared intelligence across cloud systems, causal inference will enhance policy resilience in evolving circumstances, and explainable AI will enhance operator confidence and automation. Collectively, these innovations will result in highly efficient and reliable autonomous cloud management.Altogether, the autonomous cloud management based on AI is becoming an essential paradigm. The suggested architecture offers a solid base upon which organizations can embrace the smart operation of clouds without jeopardizing their safety or efficacy, particularly because cloud infrastructure is becoming a crucial constituent of the worldwide economy.

REFERENCES

[1] Optimus: An efficient dynamic resource scheduler for deep learning clusters. Proceedings of the ACM European Conference on Computer Systems (EuroSys), 2018, 3–16.
[2] Sanusi, A.N., Bayeroju, O.F. &Nwokediegwu, Z.Q.S., 2023. Framework for Leveraging Artificial Intelligence in Monitoring Environmental Impacts of Green Buildings. International Journal of Advanced Multidisciplinary Research and Studies, 3(1), pp.1194- 1203.
[3] Mao, H., Netravali, R., & Alizadeh, M. (2017). Real experience-driven network video streaming with Pensieve. Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM), 117(4), 1–14.
[4] Guneet Kaur Walia, Mohit Kumar, and Sukhpal Singh Gill. ”AI-empowered fog/edge resource management for IoT applications: A comprehensive review, research challenges and future perspectives.” IEEE Communications Surveys & Tutorials (2023).
[5] Sijing Duan, Dan Wang, Ju Ren, Feng Lyu, Ye Zhang, Huaqing Wu, and Xuemin Shen. ”Distributed artificial intelligence empowered by end-edge-cloud computing: A survey.” IEEE Communications Surveys & Tutorials 25, no. 1 (2022): 591-624.
[6] Mohit Kumar, Atul Rai, Surbhit, and Neeraj Kumar. ”Autonomic edge cloud assisted framework for heart disease prediction using RF-LRG algorithm.” Multimedia Tools and Applications 83, no. 2 (2024): 5929-5953.
[7] (32)Victor Casamayor Pujol, Praveen Kumar Donta, Andrea Morichetta, Ilir Murturi, and Schahram Dustdar. ”Edge intelligence—research opportunities for distributed computing continuum systems.” IEEE Internet Computing 27, no. 4 (2023): 53-74.
[8] Sukhpal Singh Gill and Rajkumar Buyya. 2024. Transforming Research with Quantum Computing. Journal of Economy and Technology 2 (2024), 1–8.
[9] Jolly I. Ogbole1” A Generative AI Framework for Self-Learning and Autonomous Optimization in Cloud Infrastructure Management” (IJCSMT) E-ISSN 2545-5699 P-ISSN 2695-1924 Vol 10. No. 6, 2024.
[10] Shetty, M., Singh, A., and Patel, R.,“AI-Driven Autonomous Cloud Operations Using AIOps,”*arXiv preprint arXiv:2407.12165*, pp. 1–12, 2024.
[11] Reddy K., Kumar, S., and Rao, P., Autonomous Cloud Systems: A Survey on Self-Managing Infrastructure,”*ACM Computing Surveys*, vol. 56, no. 1, pp. 1–36, 2024.
[12] Chen, X., Zhang, Y., and Liu, H.,“AI-Powered Resource Orchestration in Cloud-Native Environments,”*IEEE Access*, vol. 12, pp. 34567–34580, 2024.
[13] Gupta, A., Verma, S., and Yadav, N.,“Machine Learning-Based Dynamic Scaling for Microservices in Kubernetes,” *Journal of Systems and Software*, vol. 210, pp. 111–125, 2024.
[14] Hassan, M., and Rahman, A., “Intelligent Cloud Operations Using AIOps: Techniques and Challenges,”*Future Internet*, vol. 16, no. 2, pp. 1–18, 2024.
[15] Das, S., Banerjee, P., and Roy, S.,“Deep Learning for Autonomous Cloud Resource Management: Recent Advances,”*IEEE Transactions on Artificial Intelligence*, pp. 1–12, 2024.

| ISSN: 2349-6037 | www.ijmserh.com ||Impact Factor: 9.274||Quarterly, Peer Reviewed & Refereed Journal|

|| Volume 12, Issue 1, January-March 2024 ||

DOI: 10.15662/IJMSERH.2024.1201052

[16] Mirhoseini, A., Goldie, A., Yazgan, M., Jiang, J. W., Songhori, E., Wang, S., Lee, Y.-J., Johnson, E., Pathak, O., Nazi, A., Pak, J., Tong, A., Srinivasa, K., Hang, W., Tuncer, E., Le, Q. V., Laudon, J., Ho, R., Carpenter, R., & Dean, J. (2021). A graph placement methodology for fast chip design. Nature, 594(7862), 207–212.
[17] Burns, B., Grant, B., Oppenheimer, D., Brewer, E., & Wilkes, J. (2016). Borg, Omega, and Kubernetes: Lessons learned from three container-management systems over a decade. ACM Queue, 14(1), 70–93.
[18] Rzadca, K., Findeisen, P., Swiderski, J., Zych, P., Bronson, P., Brokowski, D., Burczynski, C., Chaiken, R., Holler, A., & Solnica, A. (2020). Autopilot: Workload autoscaling at Google. Proceedings of the ACM European Conference on Computer Systems (EuroSys), 2020, 1–16.
[19] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. Proceedings of the IEEE International Conference on Data Mining (ICDM), 2008, 413–422.
[20] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

International Journal of Multidisciplinary and Scientific Emerging Research (IJMSERH)

Impact Factor: 9.274

ijmserh@gmail.com

www.ijmserh.com