
**E-ISSN:** 2707-6644
**P-ISSN:** 2707-6636
IJCPDM 2023; 1(1): 90-96
Received: 30-01-2023
Accepted: 07-03-2023

**Manideep Yenugula**
Kroger, Blue ash, Ohio, 45242, United State of America

# Monitoring performance computing environments and autoscaling using AI

## Manideep Yenugula

**DOI:** https://doi.org/10.33545/27076636.2023.v4.i1a.87

### Abstract
The fluctuating workloads and erratic demands faced by modern enterprises make efficient use of computer resources necessary. Cloud computing autoscaling provides an answer by allowing apps to independently modify their ability to operate in response to fluctuating demand. In order to overcome the difficulties in monitoring as well as autoscaling, this article investigates the use of AI-powered algorithms, taking into account variables including memory needs, network traffic, CPU use, and custom metrics. AI-driven models provide several benefits, such as better use of resources, scalability, dependability, lower maintenance costs, continuous availability, affordability, and easier computing environment management. On the other hand, there are several significant disadvantages, including setup complexity, possible performance deterioration, uneven performance, security issues, and higher expenses. This study assesses the contribution of AI to the resolution of issues encountered by alternative methodologies by comparing AI-powered techniques with other conventional methods. This paper illustrates the advantages of AI-driven solutions in several areas, including CPU consumption, memory usage, throughput, and reaction time, via experimental assessment and thorough analysis. Additionally, the study points out a number of areas that still need work in order to maximize effectiveness and lower computing expenses.

**Keywords:** Performance computing environment, monitoring, autoscaling, artificial intelligence (AI), workload dynamics, resource optimization, rule-based scaling, predictive scaling, threshold-based scaling, cost optimization

### Introduction
The presence of air is essential for all forms of life on our planet. Air pollution is becoming worse due to the use of non-renewable energy sources with certain industrial characteristics. Because these elements impact the health and wealth of all species on Earth, it is crucial that we keep a close eye on the state of the air we breathe at all times [1]. The need for real-time evaluation in smart applications has made cloud video surveillance a highly debated subject, coinciding with the fast growth of cloud computing, the IoT, and AI. When it comes to surveillance systems, object detection is often crucial for keeping tabs on the surroundings and activities. With the new edge-cloud computing architecture, we can handle the massive amounts of surveillance data produced continually by IoT equipment locally. But because of the complicated surveillance environment, the effectiveness of detection is still not up to scratch [2]. Among the rapidly expanding set of tools that may improve the efficiency of different systems, AI is making waves. Face recognition-based security checks can make airport security control more efficient and safe, home automation with Amazon Alexa is within reach, and AI is bolstering a number of customer service initiatives.

### Autoscaling vendors
Numerous suppliers provide autoscaling technology and services. The ability to enable autoscaling is available via services offered by each of the main public cloud providers. Frequently grouped under "cloud administration platforms," third-party services help enterprises improve their cloud installations, including autoscaling rules that connect to a cloud provider's platform.

### Several cloud service providers with autoscaling features include
- **Amazon Web Services (AWS):** AWS offers a number of autoscaling services, such as Amazon EC2 Auto Scaling and AWS Auto Scaling. Users that need to scale capacity across many AWS services may utilize AWS Auto Scaling. The goal of the Amazon

**Corresponding Author:**
**Manideep Yenugula**
Kroger, Blue ash, Ohio, 45242, United State of America

EC2 automatic scaling service, on the other hand, is to enable autoscaling for instances of Amazon EC2 that provide virtual computing resources.

- **Google Compute Engine (GCE):** GCE offers its cloud customers that operate managed instances groups of VM (virtual machine) instance the functionality of autoscaling. Groups of similar virtual machines are distributed across Google Compute Engine in a controlled manner to ensure improved availability. Controlled instance group serve as an optional installation method.
- **IBM Cloud:** For IBM Cloud workloads, IBM has a module called cluster-autoscaler that may be installed. The total amount of nodes in a group may be increased or decreased by this autoscaler in accordance with the scaling requirements specified by planned workload regulations.
- **Microsoft Azure:** The Azure AutoScale feature allows customers of the Microsoft Azure cloud service to scale resources automatically. It is possible to use VM, mobile, and website deployments with Azure AutoScale.
- **Oracle Cloud Infrastructure:** Within its cloud infrastructure system, Oracle offers several autoscaling services, such as Compute Autoscaling with an adaptable load balancer that facilitates dynamic load balancing for internet traffic.

Thanks to these AI-powered solutions, we are on the cusp of a new age of smart living. The development of the IoT with autonomous vehicles that do not need human intervention has accelerated the trend's pace in recent times [3]. In order to extract information and make decisions, High-Performance Big Data Analysis programmes deal with massive amounts of dispersed and diverse data. Computer systems that integrate AI with high-performance computing, the cloud, and IoT technologies are becoming more common in design. For better predictions, faster processing, and more efficient use of resources, it is crucial to match application and data needs with underlying hardware features [4]. Sentiment analysis may be used to examine how people or groups feel about other people, products, services, or social activities. Thanks to advances in deep learning, the abundance of information available online (particularly on social media), as well as rapid processing equipment, AI structures will eventually penetrate every aspect of human life and inspire us to think more profoundly on our own lives [5]. Cloud computing, enormous amounts of data, data centres, VR/AR, 5G, AI, the IoT, and optical fibre sensing are some of the newer sectors that have arisen in recent years. Our way of life has evolved as a result of these innovations, which have made once challenging jobs much easier to do. As an example, the artificial intelligence team at ten cent Corporation has used deep learning to find missing children by analysing childhood images. Realising cross-age facial recognition was challenging in the past. Society and all of mankind have benefited from deep learning. YouTube, Google, Alibaba, Facebook, as well as Tencent are just a few of the well-known corporations that have constructed several large-scale data centres to facilitate these new sectors [6]. Businesses now confront new problems brought forth by the Fourth Industrial Revolution. There are tremendous potential and enormous problems presented by the technological convergence of the physical, chemical, biological, and digital levels to both businesses and society as a whole. While emerging technologies like Cyber Physical Systems, the IoT, AI, and cloud computing have arrived, figuring out how to use them in a business or city is far from simple [7].

## Research Motivation

The intricacies brought forth by changing workloads, resource consumption, and efficiency bring hitherto unheard-of difficulties in contemporary performance computing settings. Optimizing performance requires effective autoscaling and monitoring strategies. Increasing computational expenses is a common outcome of inefficient monitoring and scaling systems. Scholars have acknowledged the need to use sophisticated methods that strike a compromise between high performance and cost-effectiveness to tackle these difficulties in intricate settings. The purpose of this article is to test the claim that using AI models for autoscaling and monitoring is more efficient than using conventional techniques.

In conclusion, an autoscaling resource within a cloud setting is a difficult and involved process. Algorithms that consider the following factors will be necessary to overcome these obstacles: (i) state transition overheads associated with changing the number of resources; (ii) the capacity to forecast workload accurately in the future; and (iii) the computation of the appropriate resource count necessary for the anticipated rise or fall in workload. Utilizing modeling predictive methodologies, the resource allocation strategy presented in this paper assigns or deallocates processors to the application by optimizing the application's usefulness over a restricted prediction horizon.

The document's remaining sections are organized as follows: Section II compares our work to pertinent research; Section III discusses the challenges of workload forecasting and how they relate to autoscaling; Section IV details our solution approach; Section V evaluates our algorithm empirically; and Section VI offers concluding remarks.

## Related Work

In [8], the lesson lays out a performance engineering strategy for improving the QoS of cloud, edge, and fog computing environments via the use of artificial intelligence and coupled simulation. This strategy is being created as part of the COSCO framework. The article provides an overview of AI and co-simulation, discussing their significance in optimising quality of service and performance engineering within the framework of fog computing. Optimal resource management choices may be made by combining simulated estimations with AI models, particularly DNNs. We also go over some examples of how to use DNNs as stand-ins for critical quality-of-service indicators during training, and how to leverage these models to create scheduling strategies for dynamic events in a dispersed fog setting. Using the COSCO framework, the course illustrates these principles. The effectiveness of an AI and simulation-based scheduler on a fog/cloud system is shown using a COSCO primitive for metric monitoring and simulation. As a last step, we offer AI baselines for fog management resource management difficulties.

Researchers presented a cloud-based system in [9] for the noncontact, real-time detection of walking duration and activity. The suggested system makes use of deep learning algorithms and freestanding millimeter wave radar sensors determined by the Internet of Things (IoT) to enable autonomous, free-living activity recognition and gait analysis. For training deep learning models, we employ range-Doppler maps generated from a collection of real-world in-home activities. Following a comparison of the prediction time and accuracy of other models using deep learning, the gating recurrent network approach was selected for real-time implementation because to its effective balancing of the two measures. The other two models, 2D-CNNLSTM and LSTM, were also considered. When it comes to categorising exercise routines performed at home by trained individuals, the GRU model achieves an overall accuracy of 93% and an accuracy of 86% when applied to fresh subjects. The system not only categorises the subject's walking durations and activities, but it also tracks their level of activity over time, how often they use the loo, how long they sleep, how active they are, how long they are out of the house, their present condition, and gait metrics. The fact that the individual is not compelled to wear or take any extra gadgets ensures that the system upholds privacy.

This study [10] puts forth a complete plan for modern oilfield as well as well surveillance by using cutting-edge edge computer applications, an innovative form of transmitting information technology, and internally developed algorithms for autonomous condition monitoring of SRP systems. The intention is to relieve relevant specialized staff members of routine duties such as data collection and preparation, allowing them to focus on important decisions pertaining to oilfield management. The dyno-cards generated by the electric load cells are sent to the proprietary production assistance software platform using an internal LPWAN-based data communication system. The incoming dyno cards may have their conditions detected automatically by using appropriately coded AI-algorithms, which also convert and analyse the related subsurface dynamograms.

The use of AI algorithms has been shown in research [11]. Incorporating edge computing into IoT systems is new to this study; doing so improves IoT system performance by lowering cloud load. When transmitting data at the network level, the Wi-Fi protocol is used. One way to create an edge server is using a Raspberry Pi. The overarching goal of this initiative is to forestall the occurrence of air, water, and other forms of pollution that pose serious threats to human and environmental health. City air management, transportation and traffic management, efficient electricity utilisation, and water pollution monitoring are all components of this smart city programme. A variety of sensors, including cameras, gas detectors, water quality monitors, and others, carry out this monitoring. We collect physiological data through the surrounding sensors, enter it into a database, run analyses to make informed decisions, display the results on the website, and communicate this information to the user over SMTP email.

The ideas presented in the [12] article are based on the IoT, computer vision, edge computing, and machine learning. The agricultural sector makes advantage of this synergy to track apple orchard production, with a focus on detection and data extraction for picking apples. The previously described concept creates a low-power information relay that connects battery-operated "things" to the internet in the form of a regional or global architecture by using the LoRaWAN (Low Power Wide Area Network) protocol. An edge device that runs on batteries handles pictures and data processing apart from the grid. Using specially trained weights and installing the whole YoloV4 architecture on a single-board machine (SBC) equipped with the appropriate camera seems like a plausible plan. Even in very dense and complicated surroundings, the suggested method achieves an excellent apple detection performance of up to 66.89%. These early findings demonstrate the viability of this cutting-edge computing strategy using AI and IoT.

An exhaustive literature review of AI big data analytics applied to BAMSs is provided in [13]. Load forecasting, water administration, occupancy detection, indoor environmental quality tracking, and other AI-based functions are covered. This article begins by providing an overview of current frameworks using a well-designed taxonomy. Various facets, such as the instructional procedure, physical space, computer systems, and potential use case, are examined in detail. After that, we conduct a critical conversation to determine the present difficulties. The purpose of the second section is to educate the reader on the practical uses of artificial intelligence and big data analytics. These three case studies highlight the use of AI-big analysis of information in BAMSs: energy detection of abnormalities in residential and commercial buildings; energy and performance optimization in sports facilities; and both. The study ends with some recommendations for the future and practical steps that may be taken to improve the efficiency and dependability of BAMSs in intelligent structures.

Researchers in [14] presented a unique, environmentally friendly mask with a customizable filter in (ME) 2, the Monitoring Equipment Mask Environment. The wearer's vital signs may be continuously monitored on (ME) 2 thanks to a rechargeable batteries, Bluetooth connection, and a computationally capable system-on-a-chip micro-controller. Its placement allows it to non-invasively monitor vitals like temperature, heart rate, as well as oxygen saturation. An accompanying smartphone app gives customers access to their health records using (ME) 2. In addition, Edge AI modules for computing allow for the detection of abnormal as well as early symptoms associated with potential diseases, such as those affecting the respiratory or heart systems, and the subsequent execution of predictive analyses, alerts, and recommendations. We evaluated a model based on machine learning that could differentiate between COVID-19 and seasonal flu using just vital signs in order to confirm the viability of integrated in-app Edge AI modules. We validate the very dependable performance of such a model with a 94.80% accuracy rate by creating fresh synthetic data.

The multitarget detection capabilities of smart IoT systems for real-time monitoring are the main focus of the authors' work [15]. An end-edge-cloud surveillance device is considering to use a newly developed deep neural network architecture called A-YONet, which is named by its ability to accomplish lightweight training and feature learning utilizing little computing resources. The advantages of YOLO with MTCNN are combined to create this model. An intelligent detection technique is then built using a preadjusting anchor box system and a multilayer fusion of features mechanism. Our proposed method is effective in increasing both detection precision and training efficiency, especially for multitarget recognition in smart IoT applications. Experiments and evaluations using two data sets-public and private-obtained from actual surveillance systems show this.

## Proposed Work

### System Model

To prove AI-driven monitoring as well as autoscaling performance's superiority over other approaches, the research ran a simulation of the system. To enhance monitoring and autoscaling, the simulation integrates machine learning techniques [Fig. 1] that are grounded on reinforcement training and neural networks. A task with different intensity and throughput is carried out, and the outcomes are compared. The measures are included in the performance comparison study between the AI-powered model and fixed count and rule-based scaling.

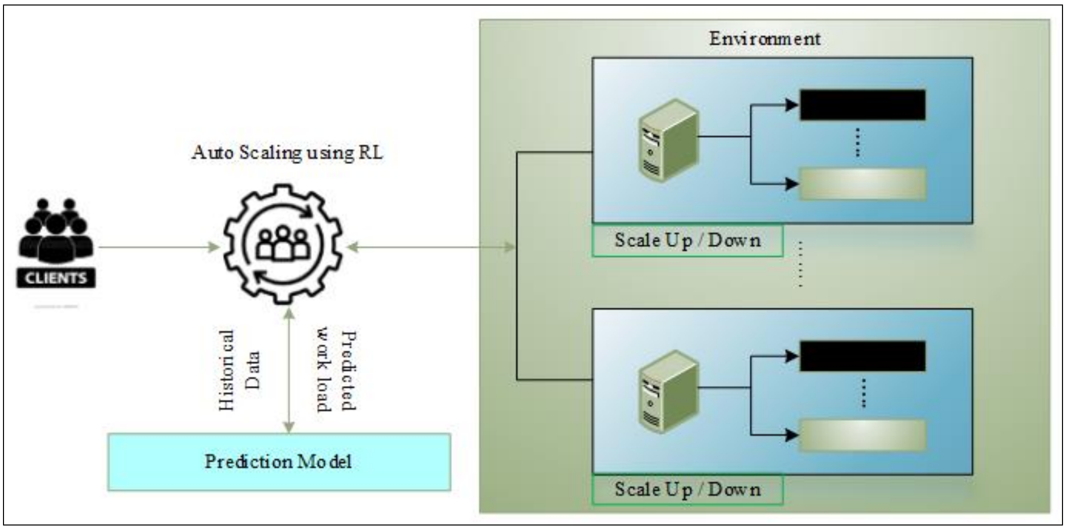


**Fig 1:** Reinforcement learning-based autoscaling framework

### RL model for Autoscaling

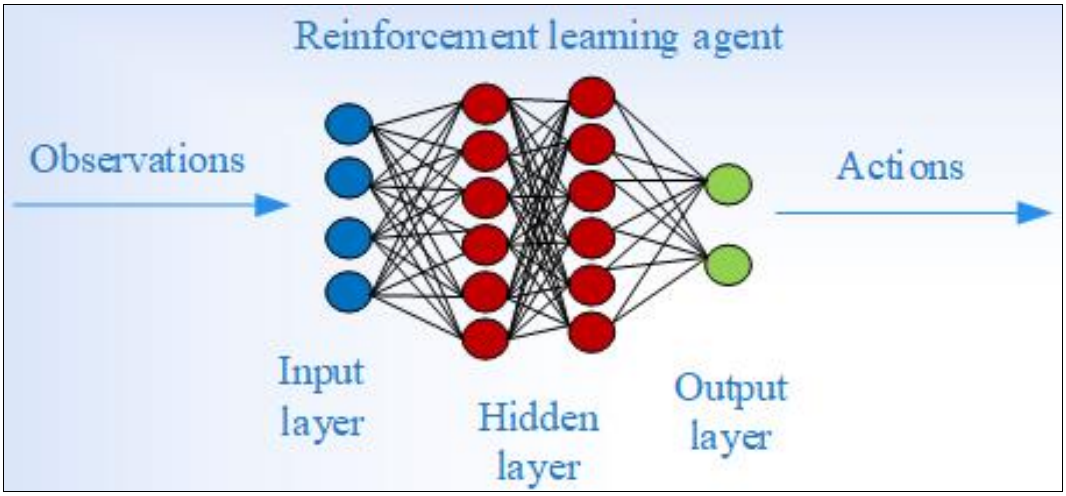


**Fig 2:** RL for Auto Scaling

It's also critical to emphasize that the following factors serve as the primary driving force behind addressing the automatic scaling issues facing apps in clouds from an RL perspective

1. Policies are dynamic, meaning that they determine an appropriate action based on the application execution and the present condition of the environment, rather than a static plan previously determined as in remedies based on meta-heuristics;
2. Policies are adaptable, meaning that online policy learning enables policy improvement as well as constant updates; and
3. Policies are transparent, meaning they do not require human involvement or deep domain knowledge. As a result, unlike rules learnt offline, learned policies are able to adjust to changes in the structure of the Cloud environment.

The input data consists of the following: the current state, the action taken, the reward earned, and the future state. The future state, or the state of the environment once the agent decides what to do, takes action, and gets a prompt response, represents the current state of the environment. With the use of this input data, reinforcement learning (RL) algorithms may solve complex problems in dynamic scenarios by iteratively interacting with the environment and optimizing cumulative benefits over time. Measurements like CPU (%), Memory utilization, Throughput (requests per minute), and Request delay (milliseconds) are collected using an application tracking agent to ascertain the current state. Seasonal trend data is also considered an input.

**Working Procedure**

1. The behavior of the learning agent at a certain moment is defined by a policy. A rule $\Pi$: $S \longrightarrow$ Mapping observed environmental conditions to appropriate actions to perform in such states is known as $A$.
2. A reward signal serves as the main justification for changing the policy as it assesses the immediate impact of the actions made in relation to the objective of the RL issue that is being confronted.
3. A value function is essential for improving policy since it outlines what is beneficial over the long run. The predicted gain, or cumulative reward over time, to be earned beginning from this state is represented by the state-value functional $V$ $(s)$. Value functions forecast future rewards associated with a given policy; the goal of value estimation is to modify policy in order to maximize reward. The environment immediately provides rewards, but values need to be (re-)estimated based on the series of experiences an agent gathers over time. There are situations when estimating the values associated with each action is necessary since the state function of values is insufficient to recommend a course of action. The projected benefit taking into account the state-action pair is represented by the action-value functional $(s, a)$.
4. The environment's model imitates the dynamic that dictates the environment's behavior. The model may be able to predict the subsequent state and reward given a state as well as an action. Models are used to make decisions about what to do by taking into account potential future circumstances before they arise (in offline mode). Nevertheless, the environment's dynamics are subject to change throughout time, and an accurate representation of the environment is not always accessible. In some instances, learning depends on real-world experiences (via the internet). Unlike the more straightforward model-free approaches, which are openly trial and error learners, model-based techniques solve RL issues using models.

Resource auto scaling enables cloud service providers running contemporary data centers to offer the greatest number of clients while guaranteeing client QoS standards in compliance with SLAs and minimizing client resource costs. To adapt resources as workloads vary, existing auto scaling systems need user interaction and API development. Reactive resource scaling causes overheads in terms of performance and also adds complexity to cloud infrastructure development. This study offers a look-ahead resource distribution method based on model predictive management that changes resources assigned to users in advance by predicting future workloads based on a constrained horizon. This approach helps to overcome these issues. Empirical findings assessing our methodology provide noteworthy advantages for Cloud service providers as well as consumers. The work shown shows that our technique is feasible when just a few machines are used. Future research will examine how scalable our methods are under the heavy workloads and many resources that characterize current applications.

The three components of the cost functional shown in Equation 5 have been utilized to assess the effectiveness of our just-in-time allocation of resources. Note that the cost of leasing a device, the cost of restructuring the app when a machine is either leased or released, and the penalty for exceeding SLA constraints are the three separate parts of the cost function. Every one of these elements has a weight associated with it. By setting the weight to an arbitrarily high number, one may force the system to always minimize a certain component. The elements of the cost functional are listed in Table 1.

$$\mathrm{Cost} = \mathrm{W_r} \times (\mathrm{R_{sla}} - \mathrm{R}) + \mathrm{W_c} \times \mathrm{M_k} + \mathrm{W_f} \times \|(\mathrm{M_k} - \mathrm{M_{k-1}})\| \quad (1)$$

**Table 1:** Components of Cost Function

| Component | Description | Unit |
|---|---|---|
| $W_r$ | Penalty for SL.A violation | S/sec |
| $W_e$ | Cost of Leasing a Machine per hour | S/machine |
| $W_f$ | Cost of reconfiguring application | S/machine |
| $R_{sia}$ | SL.A given response time | sec |
| R | Maximum response time of application | sec |
| $M_k$ | Number of machines used in the $k^{th}$ interval | Numeric |
| $M_{k-1}$ | Number of machines mesed in the k-$1^{th}$ interval | Numeric |

**Parameters:** In the MDP model, the action space was set as

$$A = \{-10, -8, \ldots, 0, \ldots, 8, 10\} \quad (2)$$

SLA "penalty" (r) =r was selected as the punishment function, and a constant c was set to at 0.1. DQN and SOR-DQN both made use of the same neural network. Due to the limited dimensionality of the state, we used a basic fully connected network with three hidden layers. Four, eight, and sixteen neurons, respectively, were selected for each of these layers. With the exception of the output layer, which used "linear" activation, all levels employed the "ReLU" activation function. Eleven neurons make up the output layer, which is equal to the total amount of activities in the MDP paradigm. Figure 1 displays a schematic design of the network.

**Results and Discussion**

Using many virtual machines (VMs) on cloud infrastructure, an efficient computing environment is built up for the experiment (Fig. 1). The virtual machines' specifications are comparable, and their memory, processor, and network bandwidth are all the same. Workloads that vary dynamically, akin to real-world or organizational scenarios, are executed by a workload generator. For autoscaling and monitoring, a variety of metrics are taken into account. In tests, Amazon AWS uses a linear auto-scaling technique that it refers to as "AWS scaling." The primary performance measure parameter is CPU usage. There are two threshold settings set up for auto-scaling: the CPU high alert threshold at 80% and the CPU cool down signal threshold at 70%. Put otherwise, the AWS dynamically raises or removes an NFV every time CPU usage rises over 80% or falls below 70%.

CPU utilization as well as memory consumption are the two indicators taken into account in performance measurements with the DC/OS system. Two threshold values-the maximum CPU utilization threshold set at 80% and the maximum memory utilization threshold set at 90%-are set up for auto-scaling.

**Table 2:** VM cost

| Scaling technique | VM time (min) | Aw SVM cost | Total Cost | % Saving |
|---|---|---|---|---|
| Fixed VM count | 43200 | $0.264/Hour | $190.08 | 0% |
| Rule based scaling | 26311 | $0.264/Hour | $115.77 | 39.1% |
| Scaling with Prediction | 25108 | $0.264/Hour | $110.48 | 42.1% |

**Table 3:** Request latency with scaling strategies

| Scaling technique | Response time P90 ms | Response time P99 ms | Response time Avg ms |
|---|---|---|---|
| Fixed VM count | 57.1 | 121.7 | 76.45 |
| Rule based scaling | 56.3 | 102.2 | 68.65 |
| Scaling with prediction | 55.9 | 96.9 | 62.23 |

The program under test was examined using a variety of metrics, and it was found that the CPU% as well as request latency were especially impacted. Different scaling strategies were used to examine the metrics CPU% as well as request delay after a workload was simulated throughout both peak as well as off peak hours. Request latencies were impacted when a particular amount of virtual machines (VMs) were being used because of either low CPU usage or a surge to a higher value (>75%). However, even with rule-based scaling, there was a brief increase in the CPU% until the virtual machine (VM) was ready to take on the load. This situation is shown in Fig. 2 shows how it led to more VM requests. This sometimes resulted in the overuse of VM resources. To fulfill future requests, virtual machines (VMs) were requested far in advance thanks to prediction-based scaling. This method offered a more effective solution by processing requests within the anticipated CPU range and latencies. Comparing the assessment of VM expenses to the use of a fixed VM count, Table 1 shows a noteworthy 39% savings.

The experimental findings demonstrate the superiority of AI-driven autoscaling and monitoring over other techniques in high-performance computing settings. But there are several places that might need better for more effective capabilities. Precisely stated performance indicators are important for accuracy assessment and monitoring. Metrics and KPIs need to match business objectives and user expectations. Examples of specific metrics include error frequencies, user satisfaction, throughput, reaction time, and resource usage. The system has to collect, analyze, and store pertinent monitoring data. Network traffic, user interactions, system logs, and performance counters are some examples of specific pertinent data points in tracking performance and autoscaling. For this architecture, a sophisticated configuration framework calls for advanced knowledge. For the IT staff, setting up AI-powered automatic scaling and tracking may be difficult and time-consuming.

## Conclusion

This work has examined the difficulties associated with autoscaling and performance computing settings, as well as the benefits and drawbacks of using RL-powered models. The research looks at capacity issues and a number of autoscaling strategies, including rule-based, fixed, and predictive autoscaling. According to the testing results, the RL-powered methodology outperformed the other techniques in terms of response time, throughput, CPU use, and memory consumption. It can be demonstrated that RL is superior.

## Future Work

Nevertheless, complexity, scalability issues, and dynamic workload fluctuations persist in RL-powered models. Future developments may concentrate on boosting data quality and prediction model accuracy to enhance policy compliance, security, and decision-making. Creating a heuristic to ascertain the significance of w for every issue occurrence would be helpful. We think that the algorithm's capacity to generalize would greatly increase with a decent heuristic for selecting w.

## References


1. Anonymous. Big Data and Artificial Intelligence Analytics in Geosciences: Promises and Potential. Last Call for 2019 Annual Meeting Proposals; c2019.
2. Chakraborty I. Toward Energy-Efficient Machine Learning: Algorithms and Analog Compute-In-Memory Hardware; c2021.
3. Vermesan O, Coppola M, Bahr R, Bellmann RO, Martinsen JE, Kristoffersen A, *et al*. An Intelligent Real-Time Edge Processing Maintenance System for Industrial Manufacturing, Control, and Diagnostic. Frontiers in Chemical Engineering; c2022.
4. Lv Z. Practical Application of Internet of Things in the Creation of Intelligent Services and Environments. Frontiers in The Internet of Things; c2022.
5. Pilato CM, Bohm S, Brocheton F, Castrillón J, Cevasco R, Cima V, *et al*. EVEREST: A design environment for extreme-scale big data analytics on heterogeneous platforms. In: 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE); c2021. p. 1320-1325.
6. Ma L, Khan NM, Cao K, Rehman H, Salman S, Rehman FU, *et al*. Prediction of Sandstone Dilatancy Point in Different Water Contents Using Infrared Radiation Characteristic: Experimental and Machine Learning Approaches. Lithosphere; c2022.
7. Echeverría-Villagómez S. MESURA 4.0 a methodology to help companies with metrology 4.0

towards industry 4.0. NCSL International Workshop & Symposium Conference Proceedings; c2019.

8. Tuli S, Casale G. Optimizing the Performance of Fog Computing Environments Using AI and Co-Simulation. Companion of the 2022 ACM/SPEC International Conference on Performance Engineering; c2022.
9. Abedi H, Ansariyan A, Morita PP, Wong A, Boger J, Shaker G, *et al*. AI-Powered Noncontact In-Home Gait Monitoring and Activity Recognition System Based on mm-Wave FMCW Radar and Cloud Computing. IEEE Internet of Things Journal. 2022;10:9465-9481.
10. Windisch C. The SMART SRP Well - Application of Edge Analytics for Automated Well Performance Control and Condition Monitoring in a Mature Brownfield Environment - A Case Study from Austria. Day 2 Wed; c2021 Nov 24.
11. Thalluri LN, Venkat SN, Prasad CV, Kumar DV, Kumar KP, Sarma AV, *et al*. Artificial Intelligence Enabled Smart City IoT System using Edge Computing. In: 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC); c2021. p. 12-20.
12. Savvidis P, Papakostas GA. Remote Crop Sensing with IoT and AI on the Edge. In: 2021 IEEE World AI IoT Congress (AIIoT); c2021. p. 0048-0054.
13. Himeur Y, Elnour M, Fadli F, Meskin N, Petri I, Rezgui Y, *et al*. AI-big data analytics for building automation and management systems: A survey, actual challenges and future perspectives. Artificial Intelligence Review. 2022;56:4929-5021.
14. Pazienza A, Monte D. Introducing the Monitoring Equipment Mask Environment. Sensors (Basel, Switzerland); c2022. p. 22.
15. Zhou X, Xu X, Liang W, Zeng Z, Yan Z. Deep-Learning-Enhanced Multitarget Detection for End-Edge-Cloud Surveillance in Smart IoT. IEEE Internet of Things Journal. 2021;8:12588-12596.