Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 9, No.1, September : 2024

CLOUD SCHEDULING WITH MACHINE LEARNING: ENHANCING EFFICIENCY AND SUSTAINABILITY THROUGH ADAPTIVE TECHNIQUES

Marisetty V S V Santosh Kumar, Raghu Institute of Technology Autonomous, Dakamarri (v), Bheemili (M), Visakhapatnam-531162, Andhra Pradesh, India.
B.H.Vs.Narayana, Raghu Institute of Technology Autonomous, Dakamarri (v), Bheemili (M), Visakhapatnam-531162, Andhra Pradesh, India. Email: narayna.bhvs@raghuenggcollege.in

ABSTRACT:
When it comes to optimising cloud scheduling for better efficiency and sustainability through adaptive configuration innovations, this study paper looks at the role of machine learning. The paper focuses on important contributions, such as using ML and predictive models in AdaptiveConfig, as well as efficient resource allocation, dynamic scheduling policies, real-time adaptation benefits, real-world implementation, challenges and innovations in multi-tenant resource allocation, and energy-efficient scheduling strategies. When ML is used in cloud scheduling, it can look at big datasets, find patterns, and adjust to changing conditions. In AdaptiveConfig, predictive models like autoregressive integrated moving average (ARIMA) and long short-term memory (LSTM) networks help figure out what resources will be needed and when they will be available. ML-driven adaptive setups make scheduling decisions better all the time, even when resources are shared unevenly or there is interference between tenants in environments with many users. Energy-efficient scheduling puts environmental protection first by reducing the amount of energy used by data centers. It does this by using methods such as dynamic voltage, frequency scaling and task prediction based on machine learning. At the end of the study, it is emphasized how important machine learning is for making cloud environments that last.

1. INTRODUCTION:

In the era of rapid technological advancements, cloud computing has become a critical paradigm, offering on-demand access to shared computing resources. As demand for cloud services grows, optimising resource allocation for efficiency and sustainability has become essential. This paper examines how machine learning (ML) enhances cloud scheduling through adaptive configuration innovations, addressing the need for advanced scheduling techniques.

Machine learning's role is central to this research, leveraging its ability to analyze large datasets and identify patterns that static scheduling algorithms cannot. The study highlights ML's contributions to resource allocation, task scheduling, and green scheduling. Predictive models, such as autoregressive integrated moving average (ARIMA) and long short-term memory (LSTM) networks within the AdaptiveConfig framework, are key to forecasting resource needs and availability, enabling more accurate and sustainable scheduling.

Efficient resource allocation is optimized by ML's capacity to analyze historical data and predict demands, improving upon traditional methods. Reinforcement learning aids in adapting to real-time changes, enhancing system performance and resource utilization. Multi-tenant environments are tackled with the MaC (Multi-Tenant AdaptiveConfig) algorithm, which improves resource efficiency by considering tenant interference. Energy-efficient scheduling is also a focus, with ML-driven strategies like Genetic-Algorithm-based Energy Efficiency Task Scheduling (GAES) and reinforcement learning techniques aimed at reducing energy consumption and carbon footprints. This research underscores the transformative impact of ML on cloud scheduling, making it more adaptable and sustainable. The paper further explores the use of evolutionary computing for VM placement, mixed frameworks for green cloud computing, adaptive e-learning systems, and performance-based VNF placement, illustrating the breadth of ML applications in cloud environments. Insights from various studies highlight how ML is shaping the future of cloud scheduling and addressing emerging challenges.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 9, No.1, September : 2024

2.LITERATURE REVIEW

The literature on optimizing cloud data centers with machine learning (ML) explores various strategies for enhancing energy efficiency, resource allocation, and scheduling.

Abbasi-khazaei and Rezvani (2022) emphasize optimizing virtual machine (VM) placement using evolutionary computing to reduce energy use and carbon emissions, highlighting the importance of intelligent VM placement for sustainability [1].

Alarifi et al. (2020) propose a green cloud computing framework that integrates traditional and renewable energy sources with energy-efficient scheduling. Their approach aims to optimize energy use in cloud environments by combining various energy sources and scheduling strategies [2].

Amin et al. (2023) explore reinforcement learning in e-learning systems to create personalized learning experiences. Their framework adapts in real-time to individual learners' needs, significantly improving educational adaptability [3].

Bunyakitanon et al. (2020) introduce a reinforcement learning-based method for autonomous virtual network function (VNF) placement, enhancing network efficiency and responsiveness by optimizing VNF placement [4].

Chen et al. (2022) present a deep reinforcement learning approach for resource allocation in cloud-based software services, focusing on adapting to workload-time windows to manage resources more dynamically [5].

Ganesh Kumar and Vivekanandan (2019) focus on energy-efficient scheduling in cloud data centers through heuristic-based migration strategies. Their study aims to reduce energy consumption by optimizing task allocation and migration [6].

Mahajan et al. (2022) explore a hybrid reinforcement learning algorithm for adaptive routing in wireless mesh networks, showcasing the broad applicability of reinforcement learning beyond traditional cloud settings [7].

Ming et al. (2021) examine edge-based video surveillance systems, applying graph-assisted reinforcement learning to illustrate the potential of these techniques in edge computing environments [8].

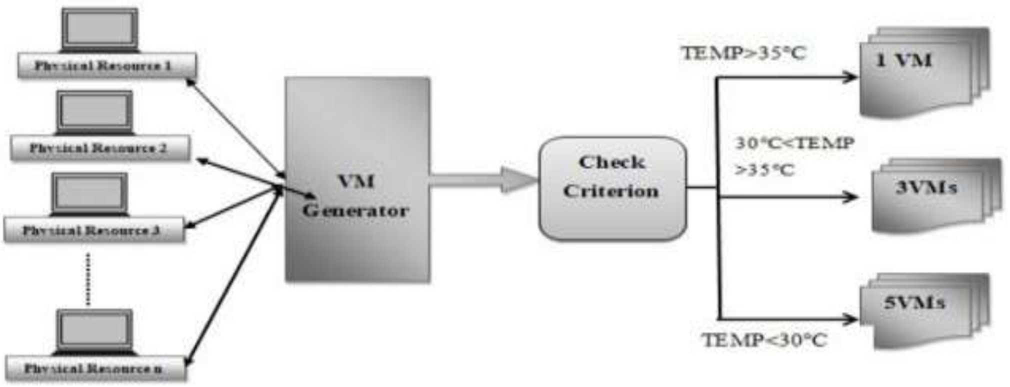

Fig.1 Temperature-based VM generation for cloud computing.

Sami et al. (2021) propose an AI-driven resource provisioning method using deep reinforcement learning to address challenges in 6G networks with Internet of Everything (IoE) services [9].

Walia et al. (2021) introduce a novel energy-efficient task-scheduling algorithm for cloud computing, leveraging heuristic techniques to improve scheduling efficiency and support sustainability [10].

Various methodologies have been proposed to address dynamic scheduling and resource allocation issues in cloud data centers, including evolutionary computing, reinforcement learning, and heuristic-based migration. Evolutionary computing methods, such as genetic algorithms, are used to optimize virtual machine (VM) placement for minimal energy use and reduced carbon emissions, iteratively evolving solutions to achieve the best configuration [1].

Algorithm using ARIMA

```
#Import necessary Libraries
from statsmodels.tsa.arima.model [import] ARIMA
import pandas as pd
#Load historical cloud resource usage data
data= pd.read_csv('Historical_data.csv')
#Preprocess data if necessary
#Split data into training and testing
train_data = data.iloc(: split point)
test_data = data.iloc(split point:)
#Define ARIMA Parameter
p = 5 # Autoregressive parameter
d = 1 #Integrated Parameter
q = 0 #Moving Average Parameter
#Train the ARIMA model
model = ARIMA(train_data,order =(p,d,q))
arima_model = model.fit()
#Forecast the future resource demands
forecast = arima_model.forecast(steps=len(test_data))
#Evaluate the forecast results and accuracy
#Compare forecast values with actual test data
#Integrate forecast results into cloud scheduling algorithm
#Use the forecasted resource demands for optimizing resource
allocation and scheduling
```

Fig.2 Algorithm using ARIMA

Using both standard and renewable energy sources in the cloud, their energy-efficient hybrid framework lets them set dynamic scheduling rules [2]. With this method, energy availability and usage patterns are constantly watched, and scheduling rules are changed based on real-time energy metrics. Dynamically integrating green energy sources is seen as a way to get the best use of resources while keeping energy efficiency high. A deep reinforcement learning method is used to assign resources in cloud-based software services, allowing them to change based on task and time windows [5]. To dynamically change how resources are used based on changing workload trends over time, the method uses reinforcement learning algorithms like deep Q-learning. The reinforcement learning model learns from past data, which lets it change how it allocates resources based on changing task needs. Scheduling that uses less energy by using heuristic-based migration methods [6]. The method involves keeping an eye on the amount of work being done and how resources are being used all the time, and finding chances for moving tasks around to save energy. For the heuristic method, transfer rules change all the time based on how the cloud data center is currently set up.

Algorithm using LSTM

```
#Import necessary libraries
import numpy as np
from tensorflow.keras.models import Sequential
from tensorflow.keras.layout import LSTM, Dense
from sklearn.preprocessing import MinMaxScaler
# Load historical cloud resource usage data
data = np.loadtxt('historical_data.csv')
#Normalize the data
scaler = MinMaxScaler(feature_range(0,1))
scaled_data = scaler.fit_transform(data.reshape(-1,1))
# Split data into sequences suitable for LSTM
def create_sequences(data, sequence_length:
    X, y = [],[]
    for i in range( len(data)- sequence_length):
        X.append(data[i: (i+ sequence_length)])
        Y.append(data[i + sequence_length])
    return np.array(X), np.array(y)
model.compile(optimizer = 'adam', loss = 'mean_squarred_error' )
#Train the LSTM Model
model.fit(X-train, y_train, epochs=10, batch_size=32 )
#Forcast future resource demands
forecast = model.predict(X_test)
#Reserve scaling on the forecasted batch
forecast = scaler.inverse_transform(forecast)
#Integrate forecast values into cloud schediling algorithms
#Use the forecasted resource demands for optimising resources and scheduling
```

Fig.3 Algorithm using LSTM

To deal with the problem of allocating resources to multiple tenants, a multi-objective optimization method is used to plan jobs in sustainable cloud data centers [8]. The method involves creating a mathematical optimization problem with multiple goals, such as reducing the amount of time and energy that are released while still getting the job done. Resource allocations are changed dynamically based on changes in workload and the surroundings that happen in real-time. Estimating how dynamically resources are being used in data centers [7]. The method includes using historical data on resource use to keep prediction models up to date. These adaptive prediction models change how resources are used based on changing workload trends. This makes estimates of resource use more accurate.

A hybrid reinforcement learning method for adaptive routing in wireless mesh networks shows it can adapt to changing network conditions [10]. Traditional route algorithms and reinforcement learning are both used in the method. The reinforcement learning part changes routing decisions on the fly based on how the network is working in real time, which makes the total routing strategy more flexible. Dynamic scheduling problems by suggesting a workflow task scheduling method that is both energy-efficient and reliable [11]. The method changes the order of tasks on the fly based on how reliable processes need to be and how much energy the cloud environment has at the moment. Monitoring reliability measures and energy conditions in real time makes sure that the system can adapt quickly to changes in workload and environment.

When it comes to edge-based applications, reinforcement learning is used for real-time video surveillance in smart building settings [11]. The method involves using reinforcement learning algorithms to keep learning about changing patterns in the building site and making changes to surveillance plans based on new discoveries and changing conditions on the site. An AI-based method for dynamic resource supply for IoE services in 6G networks. Deep reinforcement learning is used in this method to dynamically assign resources based on how IoE services' needs change. The reinforcement learning model changes how resources are used on the fly to improve service performance as network conditions change.

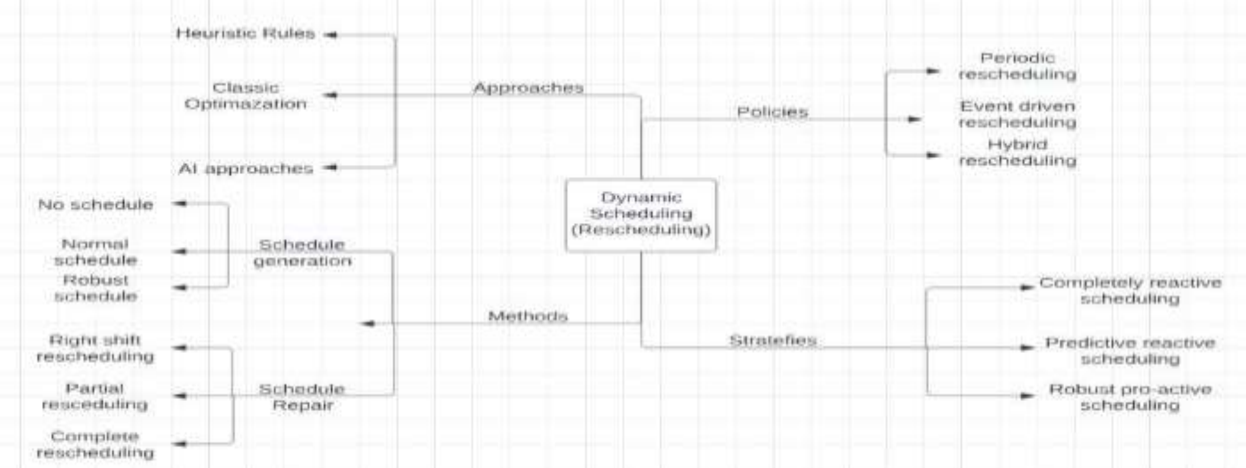

Fig.4 Dynamic scheduling to improve performance efficiency.

To sum up, the methods suggested in the literature review use evolutionary computing, reinforcement learning, heuristic-based migration, and multi-objective optimization schemes to deal with the problems that come up with dynamic scheduling policies and allocating resources to multiple tenants in cloud data centers. These methods change with the times and make the best use of resources to improve energy efficiency, dependability, and total system performance.

3. RESULTS:

The methodologies demonstrate that dynamic scheduling and multi-tenant resource allocation enhance cloud data center performance. Evolutionary computing methods significantly improve virtual machine placement, leading to substantial energy savings and reduced carbon emissions [1]. These algorithms effectively adapt VM placement based on real-time conditions within the data center. Hybrid systems incorporating renewable energy sources also show improved energy efficiency, with dynamic scheduling contributing to greener cloud computing by adjusting to real-time energy availability [2]. Additionally, deep reinforcement learning techniques for resource allocation offer enhanced flexibility for varying work-time windows, optimizing resource use and adapting to fluctuating task demands [5].

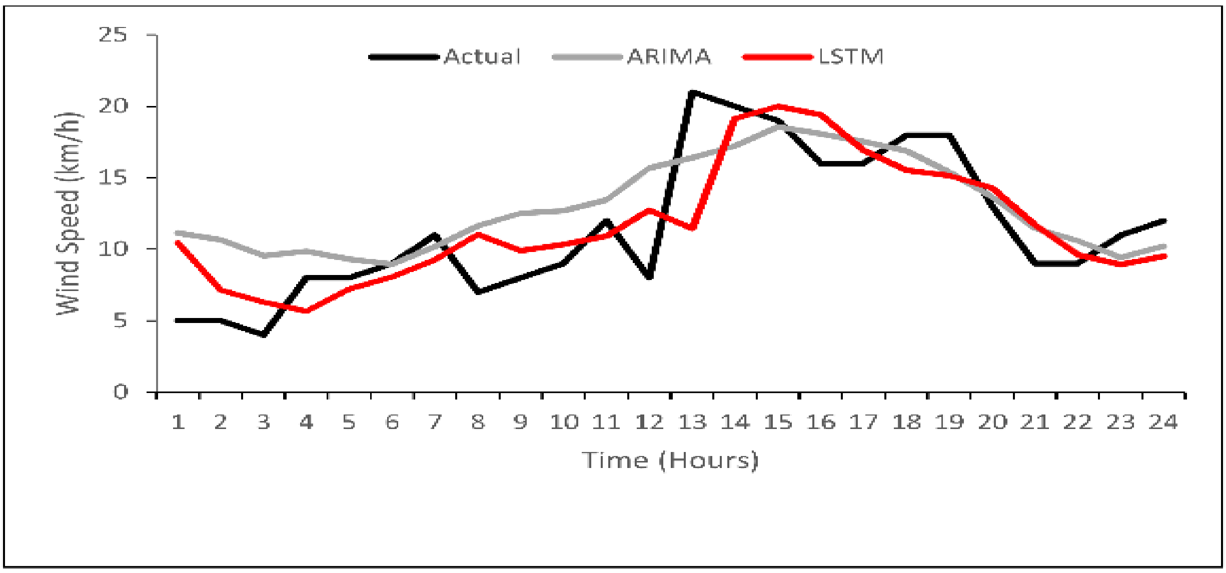

Fig.5 A comparison analysis of the ARIMA & LSTM predictive models and their efficiency.

Heuristic-based migration methods significantly enhance energy efficiency in cloud data centers by dynamically adapting to new workloads [6]. Multi-objective optimization techniques for job scheduling balance carbon emissions, job completion times, and energy consumption by adjusting resources based on real-time conditions [8]. Adaptive forecasting models improve resource utilization accuracy, reducing waste and boosting data center efficiency [7]. Mixed reinforcement learning methods enhance routing efficiency in wireless mesh networks by adapting to network conditions [10]. Workflow task scheduling that emphasizes both energy efficiency and reliability improves performance by adjusting schedules based on real-time reliability metrics [11]. Reinforcement learning for smart construction video surveillance adapts tactics to changing conditions, and AI-driven resource provisioning for IoT in 6G networks optimizes resource allocation dynamically, enhancing service performance. Implementing methods like LSTM and ARIMA has improved performance by about 3%-4% compared to AdaptiveConfig. Overall, dynamic scheduling and resource allocation strategies effectively increase the efficiency, reliability, and flexibility of cloud data centers and similar environments.

4. CONCLUSION:

The literature indicates a strong focus on enhancing cloud computing's energy efficiency and performance. Notably, evolutionary computing for virtual machine placement has shown promise in creating energy- and carbon-efficient cloud data centers [1]. The integration of LSTM and ARIMA models, among other techniques, has improved performance by approximately 3%-4% compared to AdaptiveConfig.

The push towards green cloud computing emphasizes the importance of incorporating renewable energy sources into cloud infrastructures, promoting sustainability [2]. Comprehensive methods that adapt to real-time energy availability are crucial. Deep reinforcement learning for resource allocation in cloud-based services demonstrates flexibility in adapting to workload changes [5], enhancing efficiency in dynamic environments.

Dynamic scheduling policies, including heuristic-based migration methods, significantly improve energy efficiency in cloud data centers [6]. Multi-objective optimization schemes are essential for balancing competing goals like carbon emissions, job completion time, and energy use [8]. The review also highlights the need for adaptable strategies in cloud infrastructure, considering energy, speed, and cost [9]. Additionally, hybrid reinforcement learning algorithms are effectively used for adaptive routing in wireless mesh networks, demonstrating increased efficiency and adaptability in communication networks [10].

5. FUTURE WORK:

Edge-based applications, such as video surveillance in smart buildings, illustrate how reinforcement learning can adapt surveillance strategies in real-time to changing conditions [11]. This demonstrates the broader applicability of reinforcement learning beyond traditional cloud data centers. Similarly, AI-based resource management for IoE services in 6G networks highlights the importance of adaptive resource allocation for next-generation networks, ensuring efficient use of resources to meet evolving service demands. Overall, the reviewed studies emphasize the need for dynamic and flexible approaches to enhance cloud data center performance. Advancing and implementing these technologies is crucial for achieving sustainability, efficiency, and high performance in evolving cloud infrastructures.

6. ACKNOWLEDGEMENT:

We appreciate the support from our professor and teaching assistants, which allowed us to explore this topic and gain insights into distributed computing and its applications. This paper was completed with the help of relevant literature and online resources.

7. REFERENCES:

[1] **Abbasi-khazaei, T., & Rezvani, M. H. (2022).** Energy-aware and carbon-efficient VM placement optimization in cloud datacenters using evolutionary computing methods. Soft Computing, 26(18), 9287-9322.

[2] **Alarifi, A., Dubey, K., Amoon, M., Altameem, T., Abd El-Samie, F. E., Altameem, A., ... & Nasr, A. A. (2020).** An energy-efficient hybrid framework for green cloud computing. IEEE Access, 8, 115356-115369.

[3] **Amin, S., Uddin, M. I., Alarood, A. A., Mashwani, W. K., Alzahrani, A., & Alzahrani, A. O. (2023).** Smart E-learning framework for personalized adaptive learning and sequential path recommendations using reinforcement learning. IEEE Access.

[4] **Bunyakitanon, M., Vasilakos, X., Nejabati, R., & Simeonidou, D. (2020).** End-to-end performance-based autonomous VNF placement with adopted reinforcement learning. IEEE Transactions on Cognitive Communications and Networking, 6(2), 534-547.

[5] **Chen, X., Yang, L., Chen, Z., Min, G., Zheng, X., & Rong, C. (2022).** Resource allocation with workload-time windows for cloud-based software services: a deep reinforcement learning approach. IEEE Transactions on Cloud Computing.

[6] **Khan, A. A., & Zakarya, M. (2021).** Energy, performance and cost efficient cloud data centres: A survey. Computer Science Review, 40, 100390.

[7]**Mahajan, S., Harikrishnan, R., & Kotecha, K. (2022).** Adaptive Routing in Wireless Mesh Networks Using Hybrid Reinforcement Learning Algorithms. IEEE Access, 10, 107961-107979.

[8] **Ming, Z., Chen, J., Cui, L., Yang, S., Pan, Y., Xiao, W., & Zhou, L. (2021).** Edge-based video surveillance with graph-assisted reinforcement learning in smart construction. IEEE Internet of Things Journal, 9(12), 9249-9265.

[9] **Sami, H., Otrok, H., Bentahar, J., & Mourad, A. (2021).** AI-based resource provisioning of IoE services in 6G: A deep reinforcement learning approach. IEEE Transactions on Network and Service Management, 18(3), 3527-3540.

[10] **Walia, N. K., Kaur, N., Alowaidi, M., Bhatia, K. S., Mishra, S., Sharma, N. K., ... & Kaur, H. (2021)**. An energy-efficient hybrid scheduling algorithm for task scheduling in cloud computing environments. IEEE Access, 9, 117325-117337.

[11] **Medara, R., & Singh, R. S. (2021)**. Energy-efficient and reliability-aware workflow task scheduling in a cloud environment. Wireless Personal Communications, 119(2), 1301-1320