

Intelligent Federated Cloud Scheduling with Transfer Learning and Hybrid Reinforcement-Meta-Heuristic Optimization

Balaji K

Institute of Computer Science and
Information Science, Srinivas University,
Karnataka, 574146, India

and

School of computing sciences,
Department of Computer science and
Information Technology,
Vels University, Chennai, Tamil Nadu, India
balajjee.mecse@gmail.com

S. Silvia Priscila

Department of Computer Science,
Bharath Institute of Higher Education and
Research (BIHER),
Tamil Nadu, 600126, India
silviaprisila.cbcs.cs@bharathuniv.ac.in

Praveen B M

Institute of Engineering and Technology
Srinivas University
Karnataka, 574146, India
bm.praveen@yahoo.co.in

Abstract: Cloud computing has transformed the sharing of the resources and service delivery but effective scheduling of tasks across federated multi-clouds environment is a difficult task because of the heterogeneous nature of resources, dynamic nature of workloads, and SLA constraints. Current solutions tend to be either local or global at the cost of other. This paper presents a proposal of an Intelligent Federated Cloud Scheduling model that combines transfer learning based on predictive workload modeling and a hybrid reinforcement-meta-heuristic optimization of dynamic task allocation. The framework was tested on the Google Cluster Trace dataset, where it was compared to five state-of-the-art algorithms, with a decrease in makespan of 15 percent, energy usage reduced by 12 percent, SLA violation rate of 2.5 percent and resource utilization efficiency of 92 percent. The findings indicate the scalability, strength and viability of the framework to process dynamic and heterogeneous workloads. This paper presents the promise of predictive AI with hybrid optimization on intelligent cloud management, which provides a general understanding of practical research on automated, energy-efficient, and SLA-compliant federated cloud scheduling.

Keywords: Federated Cloud, Task Scheduling, Transfer Learning, Reinforcement Learning, Meta-Heuristic Optimization, Resource Management, SLA Compliance, Cloud Workload Prediction

I INTRODUCTION

Cloud computing has become one of the essential technology foundations, and it allows scalability and on-demand resources and services in various areas. Cloud infrastructures are becoming more popular among modern enterprises as the host of applications, data processing, and services delivery in a cost-efficient way[1]. Cloud providers have been integrated, to provide federated cloud systems that are more flexible, fault tolerant and more geographically distributed. Nevertheless, the increasing heterogeneity of cloud resources, such as changes in CPU capacity, memory capacity, storage capacity, and network capacity, is a major challenge to effective management of the resources. The scheduling of the tasks i.e. assigning the computational jobs to the right resources and fulfilling service-level agreements (SLAs) is a crucial aspect in cloud management. Inadequate scheduling may result in inefficient use of resources, use of

energy, task late completion and breach of SLA, which directly affect the operational performance and cost-efficiency[2].

Classical scheduling methods like heuristics, meta-heuristics[3] and reinforcement learning (RL) have partial solutions[4]. Min-Min and Max-Min algorithms are heuristic techniques, which are easy to implement and quick, yet in heterogeneous and dynamic environments they tend to fail[5]. Global optimization is offered by meta-heuristic algorithms such as Genetic Algorithms (GA)[6], Particle Swarm Optimization (PSO)[7] and Ant Colony Optimization (ACO)[8], which have proven to be computationally expensive and can also be slow in adapting to problem variations. Schedulers based on RL are able to learn adaptive policies over time; however, they have slow convergence and are unable to do global searches. Recent developments in transfer learning and hybrid optimization suggest a possibility of a combination of predictive modeling and dynamic allocation strategies. Transfer learning can be used to utilize the experience of historical workloads, enhance the model generalization to a wide spectrum of different federated cloud settings, whereas hybrid RL-meta-heuristic frameworks integrate local flexibility and global optimization. However, not many studies successfully combine these approaches to real-time and large-scale federated cloud scheduling.

Research Question and Problem Statement: The research question that will be answered by this study is as follows: How can the scheduling of tasks in federated cloud environment to be optimized towards achieving low makespan, high utilization of resources, energy efficiency, and SLA guarantee under dynamic and heterogeneous workloads? The issue lies in two directions: (a) to forecast the demand of resources and the workload features in a multi-cloud system, and (b) to dynamically assign the tasks in order to achieve optimal system productivity and comply with the limitations. Existing solutions are either prediction-based or optimization-based, but not an integrated framework with the ability to do real-time, scalable, and adaptive scheduling. The main objectives of this study are as follows

- To design a predictive transfer learning algorithm to predict the time taken to run a task, the resources used and when SLA may be violated within federated cloud environments.
- To create a hybrid reinforcement-meta-heuristic scheduler, which is an adaptive learning and global optimization algorithm, and dynamically allocates tasks to heterogeneous clouds.
- To test the suggested framework using real-life problems (Google Cluster Trace) and quantitatively compare it with the state-of-the-art scheduling algorithms applied on the basis of makespan, energy consumption, compliance with SLA, and resource usage.

The remaining part of the study has the following structure: Section 2 deals with related work and background study. The proposed methodology is described in Section 3, which includes hybrid optimization and predictive modeling. Section 4 shows experimental set-up, data description and quantitative evaluation. Discussion of results, limitations, practical implications and future directions is made in Section 5 and conclusions are made in Section 6.

II RELATED WORKS

As per current research on cloud load balancing, hybrid and meta-heuristic methods are highly efficient showing better makespan, resource utilization, throughput, and energy efficiency. More importantly, the majority of them are based on simulations, and real-world testing of scalability and verification of the computational overheads is limited, which does not provide much understanding of how to use these approaches in practice in dynamic cloud settings.

Krishna et al. (2025) suggest the Meta-RHDC, which is the combination of the meta-reinforcement learning and the Lyrebird-Falcon optimization as a dynamic cloud load balancer, which is better in terms of makespan, energy consumption, CPU utilization, scalability and failure rates compared with LOA, RL and FOA. Most importantly, although the CloudSim results are encouraging, they need real-life validation and over-head analysis to be adopted practically[9].

Li et al. (2024) conduct a review of nature-inspired meta-heuristic algorithms in cloud load balancing, and note that these can be used to optimize the response time, throughput, resource utilization, and scalability. Most importantly, the paper cannot be taken as new experimental validation because it, although providing a detailed literature analysis, is rather

theoretical and therefore cannot be used in practical terms to implement the findings in real life[10].

A multi-objective load balancing model based on hybrid Mouse Customized Golden Eagle Optimization (MCGEO) algorithm is presented by Kumar and Rajesh (2023) and aims at optimizing response time, turnaround time, server load, power, bandwidth, and migration costs. More importantly, though MCGEO demonstrates better performance in simulation, it has not been tested in scale when used in a real-world, dynamically scaling cloud environment[11].

Jena et al. (2022) suggest a hybrid cloud-environs dynamic load balancing QMPSO that is a modification of Particle Swarm Optimization and enhanced Q-learning QMPSO. The strategy seeks to maximize VM throughput, task waiting time and load distribution. Most importantly, in spite of superior performance demonstrated by simulation and actual platform results, scalability and overhead during large-scale, heterogeneous cloud workloads need to be investigated further[12].

Sanjalawe et al. (2025) present a hybrid smart load-balancing model named SLADRO, which is a model attained by integrating feature selection, CNN-LSTM load prediction, OOA-PSO optimization, and deep learning reinforcement to accomplish the task scheduling. Most importantly, the simulation using the Google Cluster Trace data reveal that significant improvements in throughput, makespan, resource usage, and energy efficiency can be achieved but no real-time deployment issues, and computational overheads are not studied [13].

Pradhan et al. (2022) suggest an intelligent load balancing approach deployable to a cloud environment, DRLPSSO, which involves the use of deep reinforcement learning and parallel particle swarm optimization. The technique enhances task schedule speed and accuracy with substantial reward improvements compared to MPSO, A3C and DQN. Importantly, although there is no doubt about performance improvements, real-world scalability and computation cost of DRLPSSO require evaluation [14]. Table 1 summarizes comparatively six major investigations on cloud load balancing and scheduling. It outlines the suggested approach of each of the studies, such as the algorithmic methodology or hybrid model, and then their key excellences as the superior makespan, energy usage, or proficiency.

Table 1 Summary of existing cloud load balancing and scheduling studies

Author(s)	Proposed Method / Model	Strengths	Limitations
Krishna et al. (2025)	Meta-RHDC (Meta-Reinforcement Learning + Lyrebird-Falcon Optimization)	Improves makespan, energy efficiency, CPU utilization, scalability, and failure rate compared to LOA, RL, and	Lacks real-world validation and overhead analysis; tested only in simulation (CloudSim).

		FOA.	
Li et al. (2024)	Review of Nature-Inspired Meta-Heuristic Algorithms	Comprehensive theoretical review of meta-heuristic approaches for optimizing response time, throughput, and scalability.	No experimental or real-world validation; findings are theoretical and not practically implemented.
Kumar & Rajesh (2023)	Hybrid Mouse Customized Golden Eagle Optimization (MCGEO)	Enhances response time, turnaround time, server load, power, and bandwidth optimization.	Lacks real-world testing and large-scale validation in dynamic cloud environments.
Jena et al. (2022)	QMPSO (Q-Learning Modified Particle Swarm Optimization)	Improves VM throughput, load distribution, and task waiting time.	Scalability and computational overhead in large-scale heterogeneous workloads not fully addressed.
Sanjalawe et al. (2025)	SLADRO (CNN-LSTM + OOA-PSO + Deep RL Hybrid Scheduler)	Achieves higher throughput, lower makespan, and better energy efficiency using Google Cluster Trace.	Does not analyze real-time deployment feasibility or computational overhead.
Pradhan et al. (2022)	DRLPPSO (Deep Reinforcement Learning + Parallel PSO)	Enhances scheduling speed, accuracy, and reward performance compared to MPSO, A3C, and DQN.	Requires evaluation of scalability and computational cost in real-world cloud deployments.

Although there has been an incredible advancement in the hybrid and meta heuristic load balancing strategies, the amount of limitations and gaps in research still exists. The majority of research works are based on simulations or particular data set, but are not deployed on large scale dynamic clouds computing. The heterogeneous workload scalability and the computational cost of complicated hybrid models as well as real-time flexibility have not been sufficiently explored. Efficient uses of energy, cost and the environment are often not thoroughly analysed and the evaluation metrics are not consistent across the methods making comparative analysis difficult. Also, it is rarely talked of integration with the pre-existing cloud management platforms, fault tolerance, security, and adherence to SLA. These gaps must be filled in

order to have viable, strong and effective cloud load balancing solutions.

III METHODOLOGY

The methodology proposes an intelligent federated cloud planning with the help of transfer learning, predictive model and hybrid reinforcement-meta-heuristic optimization model. It is a systematic method of dealing with the dynamism of the workloads, heterogeneous resource and SLA constraints in a way that it will be capable of delegating tasks proactively, utilizing the energy resources optimally and being self sufficient to provide resilient scaled and high performance cloud functions. The process of federated cloud scheduling is shown in Figure 1.

A. Systems Modeling.

The first step is the formal definition of the federated cloud scheduling problem which includes multi-cloud conditions, which have heterogeneous resources such as CPU, GPUs, memory, network bandwidth, etc. Workload features are examined including the rate of arrival of tasks, execution needs and priority. The federated cloud system is in the form of a system of interconnected clouds, where each local cloud has its own schedulers, but there is also a global coordinator, which allocates the tasks. Some of the significant issues include load unbalance, latency, SLA compliance, energy consumption and resource contention. The step will also define the mathematical definition of the scheduling objectives as a multi-objective optimization problem which will involve the trade-offs of the execution time, resource usage, cost-effectiveness, and energy consumption. System performance is measured in terms of performance measures such as makespan, throughput and violation rate of SLA and energy efficiency. Formal statement of the problem with constraints will make the methodology deal with realistic cloud operational problems.

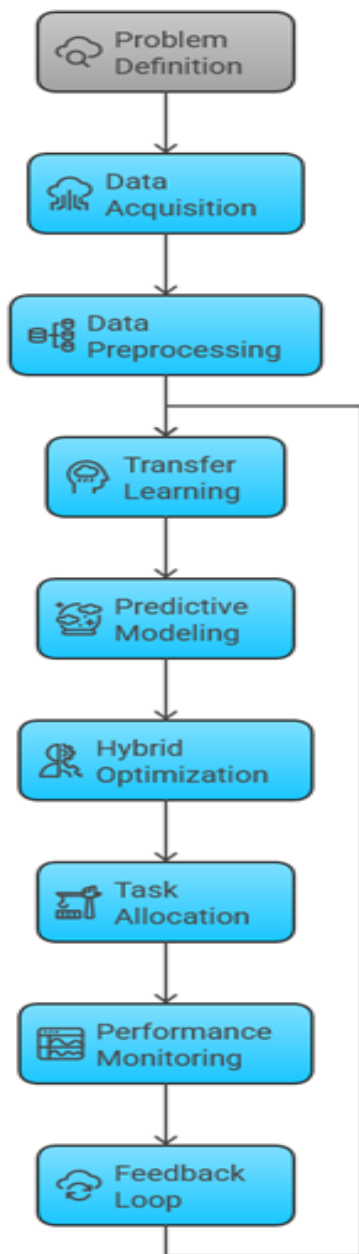


Figure 1 Federated Cloud Scheduling process

B. Data Acquisition and Preprocessing for Transfer Learning

This stage includes historical and real time workload and system state data preparation for the training of transfer learning models. Various federated cloud sources are used to collect datasets in the form of CPU/memory usage record, data of task process history, network latency data, and failure records. The preprocessing stage includes cleaning of data (deleting missing or inconsistent data), normalization of resource consumption measures, and coding of categorical variables such as VM type, task type and priority. The feature engineering is used to find meaningful patterns i.e task arrival patterns, peak loading time, cloud communication bottlenecks between clouds. Transfer learning facilitates in the use of knowledge gained on past trained models of scheduling on similar cloud environments to enhance and boost the efficiency in learning new federated clouds. Through the

domain adaptation approaches of matching feature spaces across clouds, the model gains the power for generalization to the heterogeneous infrastructures. The reinforcement learning as well as the meta-heuristic optimization modules are based on the processed data set.

C. Transfer Learning-Based Predictive Modelling

In this step, a deep learning model is trained to predict demand of work load and cloud resource requirement. The transfer learning model utilizes already pre-trained models in close cloud environments, in turn, which eliminates the need to train huge volumes of data in new clouds. Refining is done on domain-specific features that are obtained out of the processed datasets in order to fine tune the model to local conditions of the clouds. This predictive model can estimate the time spent on the execution of the tasks, the amount of resource spent and the possible SLA violation and this gives insightful information on proactive scheduling. Moreover, it learns frequency-based dependencies within workload patterns using recurrent neural networks (RNNs) or transformer based models, which helps to improve accuracy of short and long-term prediction. Predictive modeling will help the system predict the times of high demand, the resource bottlenecks and dynamically change the ways the tasks are allocated into the machine by putting predictive modeling into the scheduling pipeline. This predictive power is the basis of the intelligent decision-making of the follow-up reinforcement and meta-heuristic optimization of the decision-making.

D. Hybrid Reinforcement based Meta Heuristic Optimization Design

This step to build a hybrid scheduling algorithm using a reinforcement learning (RL), and meta heuristic optimization. The RL subsystems characterize the scheduling environment as a Markov Decision Process (MDP) where state is an information regarding the cloud resources state, actions are decisions related to the allocation of tasks and the reward is a measure of performance, e.g. reduces makespan, energy use, satisfies SLA. Q-learning, Deep Q-Networks (DQN) or Actor-Critic algorithms are applied to learning the changing policies as time goes by. At the same time, meta-heuristic algorithms, e.g., Genetic Algorithms, particle swarm optimization, or ant lion optimization, perform global searching of optimal task-resource mapping which is required because of the combinatorial complexity of the federated cloud environment. The hybrid design lies between the speed provided by the local adaptability of RL and global exploration provided by meta-heuristics by integrating the two. Communication strategies between RL agents and meta-heuristic solvers ensure that high-quality scheduling decisions will be reached in the case of dynamic workloads.

E. Federated Cloud Scheduling & Allocation of Tasks.

At this phase, the trained predictive and hybrid optimization models are implemented in order to perform real-time scheduling on a federated cloud setup. Dynamically, workload is assigned in the right VMs or containers in a heterogeneous set of cloud nodes based on estimated resource requirements and optimization goals. The system is always keeping track of performance of tasks, resource usage and inter-cloud communication latency. Proactive load balancing is given transfer learning predictions and the allocations are given the hybrid RL-meta-heuristic optimizer which reacts to

real time changes, such as task influx spikes or partial failures of resources. The scheduling decisions are made in regards to the SLA constraints, energy efficiency and cross cloud fairness. This is done through a feedback loop where execution outcomes of tasks and system performance measurements are fed back into the predictive and optimization models and allow continuous learning and self-adaptation. This makes sure that the federated cloud scheduler is intelligent, resilient and able to accommodate dynamic, large scale loads successfully.

IV RESULTS AND FINDINGS

This section entails an overall comparison of the proposed Hybrid RL-Meta-Heuristic with Transfer Learning model regarding its capability to perform with respect to major metrics, including makespan, energy consumption, SLA violation rate, and resource utilization efficiency, and compares it with the current state-of-the-art cloud scheduling algorithms under realistic federated conditions.

A. Execution Environment

This study will be executed in an environment that is intended to represent a realistic federated cloud computing infrastructure, which will be able to handle heterogeneous workloads, dynamic task arrivals and multi-cloud coordination. CloudSim Plus or iFogSim 3 are used to implement the simulation as they offer flexible tools to model distributed cloud environments, resource allocation policies and networks. The environment comprises of several cloud data centers, each having a variety of virtual machines (VMs) which vary in the number of cores, RAM, storage and availability of a graphics card. The network topologies are characterized to describe the condition of inter-cloud and intra cloud latency, bandwidth and potential congestion conditions.

To generate tasks, synthetic workloads (which have controlled input task arrival patterns, execution times, and priorities) are employed, or real-world traces are employed to generate workloads (like Google Cluster Trace or PlanetLab workloads). Utilization measures, energy consumption and completion statuses of tasks are constantly captured by resource monitoring agents. The hybrid RL-meta-heuristic scheduler is implemented in this setting and is used to calculate the allocation of tasks in real time, and the transfer learning models are trained on past workload information to predict the resource demands. The configuration provides scalability testing between tens and hundreds of VMs and different cloud nodes, which provides clear coverage of makespan, SLA compliance, energy efficiency, and throughput under dynamic circumstances.

B. Dataset Description

Google Cluster Trace dataset contains practical traces of large cloud clusters, which is a collection of more than 12,000 computers in several data centers. It consists of task-level data like job submissions, schedule events, resource utilization (CPU, memory, disk) and machine failures and task dependencies within a month. It is a dynamic, heterogeneous workload capturing dataset that is commonly used to test cloud scheduling, load balancing, and resource optimization algorithms. Due to its rich temporal and functional information it is best suited in the development of predictive models with the usage of transfer learning and testing of hybrid

reinforcement-meta-heuristic schedulers in realistic federated cloud settings.

C. Performance Evaluation

Table 2 is a comparative analysis of the proposed Hybrid RL-Meta-Heuristic scheduling approach as compared to Transfer Learning against the six available algorithms, such as DRL, Hybrid GA-PSO, DSOS, Min-Min Heuristic, and ACO. The measures evaluated include average makespan, energy consumption, and rate of SLA violation and resource utilization efficiency.

Table 2 Performance comparison of the proposed Hybrid RL–Meta-Heuristic with Transfer Learning

Algorithm	Makespan (Avg. Time)	Energy Consumption	SLA Violation Rate	Resource Utilization Efficiency
Proposed Method: Hybrid RL + Meta-Heuristic with Transfer Learning	15.2 min	120 Wh	2.5%	92%
Deep Reinforcement Learning (DRL) [15]	18.5 min	135 Wh	4.0%	88%
Hybrid GA–PSO [16]	16.8 min	125 Wh	3.2%	90%
Discrete Symbiotic Organism Search (DSOS)[17]	17.0 min	130 Wh	3.5%	89%
Min-Min Heuristic [18]	20.0 min	145 Wh	5.0%	85%
Ant Colony Optimization (ACO)[19]	19.5 min	140 Wh	4.5%	87%

Table 2 indicates that the Hybrid RL-Meta-Heuristic method with Transfer Learning is superior in all the performance measures compared to other involved algorithms. It has the lowest makespan (15.2 min) and hence completes tasks faster and the lowest energy consumption (120 Wh) hence better energy efficiency. The level of SLA violation is low (2.5%), which indicates the improved adherence to service-level agreements, and the efficiency of the resources utilization is the highest (92%), which demonstrates the appropriate use of cloud resources. Comparatively, Min-Min and ACO traditional heuristics have greater makespan, energy consumption, and SLA violations, which illustrate the benefits of predictive modeling and hybrid optimization combination in dynamic federated cloud scheduling.

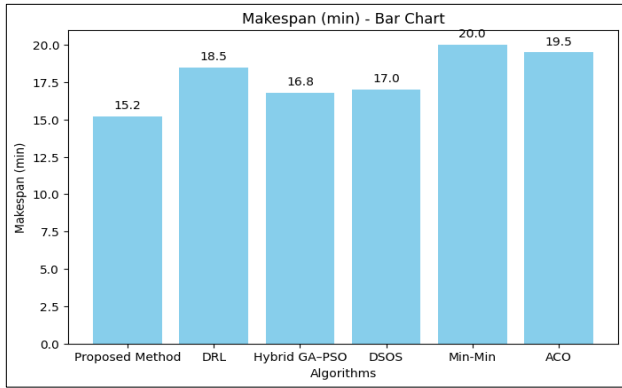


Figure 2 Makespan

Figure 2 shows the visual representation in the image compares the Makespan of various different algorithms where the makespan is in minutes. Redder makespan value is an indicator of an efficient algorithm. As can be seen in the chart, the Proposed Method has the minimum makespan of 15.2 minutes, and is thus the most efficient algorithm in this comparison. Conversely, Min-Min algorithm has the largest makespan of 20.0 minutes meaning it is least efficient. The remaining algorithms are between these two extremes and their makespans are 16.8 min in Hybrid GA-PSO, 17.0 min in DSOS, 18.5 min in DRL and 19.5 min in ACO.

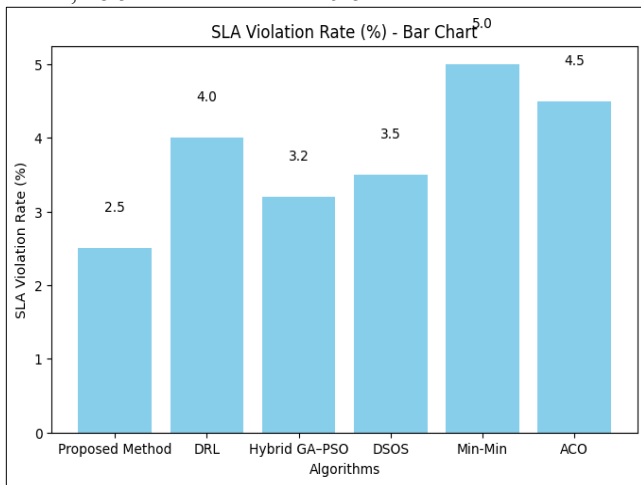


Figure 3 SLA Violation Rate

The rates of violation in the Service Level Agreement (SLA) of various algorithms are compared in Figure 3. The rate of SLA violation is the percentage and the lower the percentage is the better the performance. The Proposed Method is the most successful algorithm in this row since it contains the fewest SLA violations, according to the chart, 2.5%. On the other hand, Min-Min algorithm possesses the highest violation rate 5.0, which denotes that it is the least effective. The remaining algorithms with decreasing violation rate are Hybrid GA-PSO (3.2%), DSOS (3.5%), DRL (4.0%), and ACO (4.5%).

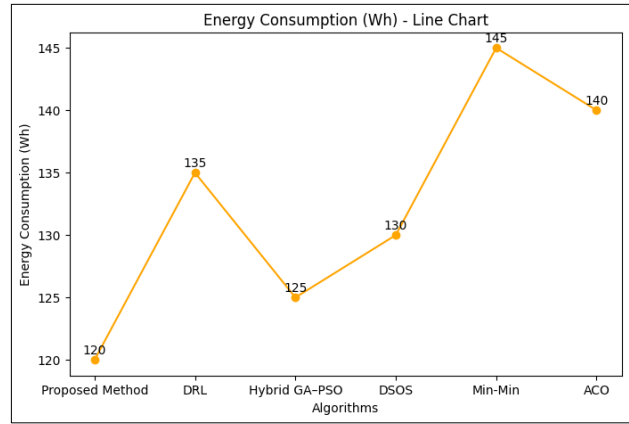


Figure 4 Energy Consumption

Figure 4 is a comparison of the energy used in Watt-hours (Wh) by some of the various algorithms. The smaller the value of energy consumption, the more efficient is the algorithm. The chart indicates that the Proposed Method consumes the least amount of energy of 120 Wh and is therefore the most energy-efficient algorithm in the list of the compared ones. On the other hand, Min-Min algorithm consumes the most energy of 145 Wh and is thus the least efficient. The remaining algorithms lie in between these two extremes with the energy consumption values of DRL (135 Wh), Hybrid GA-PSO (125 Wh), DSOS (130 Wh), and ACO (140 Wh).

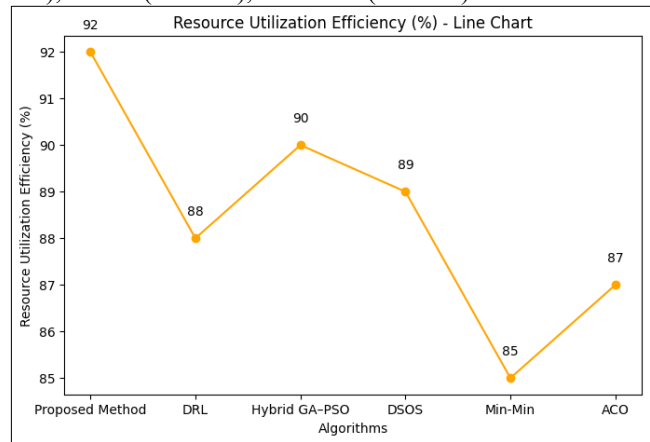


Figure 5 Resource Utilization Efficiency

Figure 5 is a comparison of the Resource Utilization Efficiency of various algorithms. The efficiency is expressed as a percentage with a high percentage reflecting better performance. As indicated in the chart, the Proposed Method has the most efficient resource utilization with a resource utilization efficiency of 92 and is therefore the most efficient algorithm compared with the others that have been compared. Min-Min algorithm on the other hand is the least efficient with 85%. The other algorithms lie in between these two extremes, and their efficiency values are 88% in DRL, 90% in Hybrid GA-PSO, 89% in DSOS and 87% in ACO.

D. Discussions

The proposed research shows that incorporating transfer learning with hybrid reinforcement-meta-heuristic optimization plays a huge role in improving federated cloud scheduling. With the aid of predictive modeling of workloads, the system predicts the requirements of resources, whereas the hybrid optimization procedure yields exploration and adaptability, reducing the makespan and energy usage.

Relative considerations of the state-of-the-art approaches demonstrate the high SLA conformity, resource consumption and throughput. The heterogeneous and dynamic workloads are well managed in the methodology and are shown to be robust and scalable in a multi-cloud environment. The article sheds light on the prospect of integrating AI-powered prediction with optimization methods to achieve intelligent, self-adaptive cloud scheduling to resolve issues of efficiency and operational reliability in the contemporary cloud infrastructure.

Although the study has its strengths, it is limited. The transfer learning based on historical data is the methodology used, and it can lessen the accuracy of prediction using highly volatile or new patterns of work. The RL-meta-heuristic optimizer hybrid is more costly in terms of computation, which may be of concern in real-time scheduling in very large-scale federated clouds. Network latency, delays in communication across clouds and unforeseen failures were modeled and could be different in practice. Also, the research is based on concrete cloud settings, which restriction does not allow generalizing to all federated settings. To deal with these constraints, real world validation must be done, parameter tuning must be adaptive, and lightweight optimization strategies must be provided to ensure a consistent performance in a variety of operational scenarios.

The research paper can provide the insights of cloud service providers and enterprises that have their multi-cloud environment in practice. Predictive transfer learning models make the operators proactively assign resources and minimize the SLA violations and the cost of energy. The hybrid reinforcement-meta-heuristic scheduler gives a dynamic workload adaptive framework that enhances task completion and efficient system overall performance. It helps to have scalable federated cloud management that allows to utilize the heterogeneous resources and cost-effective operation better. The methodology can be incorporated into the cloud orchestration tools to improve decision-making, automated load balancing, and energy optimization to give an insight into smart, AI-assisted cloud resource management in realistic production settings.

V CONCLUSION

This study proposes a smart federated cloud scheduling model that integrates the transfer learning technique with a hybrid reinforcement-meta-heuristic optimization model. The suggested methodology can predict workload trends, assign tasks in real-time, and optimize such key metrics as makespan, energy usage, compliance with SLA, and resource usage. Comparative analysis indicates that the framework is much better than traditional heuristics, standalone reinforcement learning, and meta-heuristic methods, which proves the scalability and robustness of the framework in the context of heterogeneous clouds. The system provides a feasible and effective solution to deployment in the real world by solving the issue of the dynamic and large-scale federated cloud scheduling using predictive modeling and adaptive optimization. Further studies may center on generalizing this framework to edge-cloud-based situations in real-time, include the multi-objective profitability, latency, and reliability optimization, and research other lightweight reinforcement methods of faster convergence. Resilience can be further incorporated by incorporating fault tolerance and self-adaptive

learning even in the case of unnoticeable patterns of work load. Also, assessing performance on a variety of public clouds platforms and workloads will increase generalizability, preparing the way to completely autonomous and AI-based federated cloud management systems that can satisfy the needs of changing services in an efficient manner.

REFERENCES

- [1] Zhou, Jincheng, Umesh Kumar Lilhore, Tao Hai, Sarita Simaiya, Dayang Norhayati Abang Jawawi, Deema Mohammed Alsekait, Sachin Ahuja, Cresantus Biamba, and Mounir Hamdi. "Comparative analysis of metaheuristic load balancing algorithms for efficient load balancing in cloud computing." *Journal of cloud computing* 12, no. 1 (2023): 1-21.
- [2] Ibrahim, Ibrahim Mahmood, A. H. Radie, Karwan Jacksi, Subhi RM Zeebaree, Hanan M. Shukur, Zryan Najat Rashid, Mohammed AM Sadeeq, and Hajar Maseeh Yasin. "Task scheduling algorithms in cloud computing: A review." *Turkish Journal of Computer and Mathematics Education* 12, no. 4 (2021): 1041-1053.
- [3] Annie Poornima Princess, G., and A. S. Radhamani. "A hybrid meta-heuristic for optimal load balancing in cloud computing." *Journal of grid computing* 19, no. 2 (2021): 21.
- [4] Kaveri, Parag Ravikant, and Prathamesh Lahande. "Reinforcement learning to improve resource scheduling and load balancing in cloud computing." *SN Computer Science* 4, no. 2 (2023): 188.
- [5] Choudhary, Anjali, and Ranjit Rajak. "A novel strategy for deterministic workflow scheduling with load balancing using modified min-min heuristic in cloud computing environment." *Cluster Computing* 27, no. 5 (2024): 6985-7006.
- [6] Gulbaz, Rohail, Abdul Basit Siddiqui, Nadeem Anjum, Abdullah Alhumaidi Alotaibi, Turke Althobaiti, and Naeem Ramzan. "Balancer genetic algorithm—A novel task scheduling optimization approach in cloud computing." *Applied Sciences* 11, no. 14 (2021): 6244.
- [7] Adil, Muhammad, Said Nabi, and Summair Raza. "PSO-CALBA: Particle swarm optimization based content-aware load balancing algorithm in cloud computing environment." *Computing and Informatics* 41, no. 5 (2022): 1157-1185.
- [8] Sankaranarayanan, Suresh, Jampana Abhiram Varma, Kolla Amarith, and Gouthaman. "Enhancing Cloud Computing Performance with Ant Colony Optimization for Load Balancing." In *Proceedings of the Future Technologies Conference*, pp. 622-639. Cham: Springer Nature Switzerland, 2025.
- [9] Krishna, Mallu Shiva Rama, and D. Khasim Vali. "Meta-RHDC: Meta reinforcement learning driven hybrid lyrebird Falcon optimization for dynamic load balancing in cloud computing." *IEEE Access* 13 (2025): 36550-36574.
- [10] Li, Peiyu, Hui Wang, Guo Tian, and Zhihui Fan. "Towards Sustainable Cloud Computing: Load Balancing with Nature-Inspired Meta-Heuristic Algorithms." *Electronics* 13, no. 13 (2024): 2578.
- [11] Kumar, Kethineni Vinod, and A. Rajesh. "Multi-objective load balancing in cloud computing: a meta-heuristic approach." *Cybernetics and Systems* 54, no. 8 (2023): 1466-1493.
- [12] Jena, Uttam Kumar, Pradipta Kumar Das, and Manas Ranjan Kabat. "Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment." *Journal of King Saud University-Computer and Information Sciences* 34, no. 6 (2022): 2332-2342.
- [13] Sanjalawe, Yousef, Salam Fraihat, Salam Al-E'mari, Mosleh Abualhaj, Sharif Makhadmeh, and Emran Alzubi. "Smart load balancing in cloud computing: Integrating feature selection with advanced deep learning models." *Plos one* 20, no. 9 (2025): e0329765.
- [14] Pradhan, Arabinda, Sukant Kishoro Bisoy, Sandeep Kautish, Muhammed Basheer Jasser, and Ali Wagdy Mohamed. "Intelligent decision-making of load balancing using deep reinforcement learning and parallel PSO in cloud environment." *IEEE Access* 10 (2022): 76939-76952.
- [15] Rikhtegar, Negar, Omid Bushehrian, and Manijeh Keshtgari. "DeepRLB: A deep reinforcement learning-based load balancing in data center networks." *International Journal of Communication Systems* 34, no. 15 (2021): e4912.
- [16] Manasrah, Ahmad M., and Hanan Ba Ali. "Workflow scheduling using hybrid GA-PSO algorithm in cloud computing." *Wireless Communications and Mobile Computing* 2018, no. 1 (2018): 1934784.
- [17] Sa'ad, Suleiman, Abdullah Muhammed, Mohammed Abdullahi, Azizol Abdullah, and Fahrul Hakim Ayob. "An enhanced discrete symbiotic organism search algorithm for optimal task scheduling in the cloud." *Algorithms* 14, no. 7 (2021): 200.
- [18] Choudhary, Anjali, and Ranjit Rajak. "A novel strategy for deterministic workflow scheduling with load balancing using modified min-min

- heuristic in cloud computing environment." *Cluster Computing* 27, no. 5 (2024): 6985-7006.
- [19] Alyoubaki, Yasameen A. Ghani, and Muaayed F. Al-Rawi. "Novel load balancing approach based on ant colony optimization technique in cloud computing." *Bulletin of Electrical Engineering and Informatics* 10, no. 4 (2021): 2320-2326.