Proceedings of the International Conference on Electronics and Renewable Systems (ICEARS-2026)
DVD Part Number: CFP26AV8-DVD; ISBN: 979-8-3315-4880-3

Deep Reinforcement and Meta-Heuristic Hybrid Models for Intelligent Cloud Load Distribution

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 become the backbone of the modern digital infrastructure, making the provision of on-demand, scalable and cost-efficient computing services possible. However, efficient load distribution is a major challenge because of unpredictable workloads and dynamic resource availability and service level agreement (SLA) limitations. Traditional static or heuristic-based schedulers cannot adapt well to a highly-dynamic nature of cloud environments. This study proposes a Deep Reinforcement and Meta-Heuristic Hybrid Framework (DRMHF) aimed at getting intelligent, adaptive and energy efficient cloud load distribution. The framework combines adaptive decision making through Deep Reinforcement Learning (DRL) and global search refinement through meta-heuristic optimization technique, Ant Lion Optimization (ALO). Experiments performed on CloudSim Plus based on Google Cluster workload traces show significant performance improvements compared to the existing schedulers. The proposed DRMHF achieves a makepan of 1250 s, throughput of 820 task/s, energy consumption of 52.4 kWh and only 1.8% SLA violation, which outperforms state-of-the-art approaches. These results validate the hypothesis that hybrid intelligence can play an important role in improving adaptability, scalability, and sustainability in cloud load management, and determine DRMHF as a promising strategy for autonomous resource allocation for large-scale cloud infrastructures.

Keywords: Cloud Load Balancing; Deep Reinforcement Learning; Meta-Heuristic Optimization; Hybrid Scheduling; Resource Management; Energy Efficiency; SLA Compliance; Adaptive Cloud Systems

I INTRODUCTION

One of the biggest technological advances to hit the internet over the past several years is cloud computing, which provides on-demand, flexible access to computing resources, storage resources, and applications over the internet. It supports different domains like artificial intelligence, data analytics, e-commerce and healthcare by providing cost-effective and scalable infrastructure[1]. With the ever-explosive increase in user demands and the surge of distributed services, efficient load balancing has become vitally important in order to ensure system performance, reliability, and user satisfaction. Load balancing is the technique of distributing the computational tasks optimally across the virtual machines (VMs) so that the resources never get overloaded and the latency gets less. With the proliferation of more heterogeneous and dynamic cloud environments, traditional scheduling schemes like Round Robin and Min-Min fail to scale-up the cloud environment's workload and resource changes efficiently. This has driven the adoption of intelligent computational techniques that are capable of managing resources autonomy and in real-time[2]

Recent developments in Artificial Intelligence (AI) and Machine Learning (ML) have made it possible to support intelligent decision-making in cloud computing systems [3]. Reinforcement Learning (RL)[4] and specifically Deep Reinforcement Learning (DRL)[5] has demonstrated tremendous potential for adaptive resource scheduling because of its capability of learning the best action by interacting with the environment. DRL-based schedulers are dynamic schedulers that adjust task allocation policies dynamically, so as to keep makespan low and throughput high without the knowledge of workload patterns. However, the convergence of DRL models is unstable and easy to fall in local optima, especially in high-dimensional search space[6]..

The performances of the meta-heuristic algorithms such as Particle Swarm Optimization (PSO)[7], Genetic Algorithm (GA)[8] and ALO[9] have shown strong global optimization performance. These methods cover the search space in a global manner but insufficient smartness to adapt to the evolving environment. Therefore, researchers have started to investigate hybrid models that enhance the flexibility of DRL with the exploration strength of meta-heuristics. Such integration has global and local optimizations advantages, which enhance the accuracy of load distribution, resource utilization, and energy efficiency.

Research Question and Problem Statement: Despite these advancements, there are still several challenges. Traditional and AI-based approaches find it extremely difficult to achieve a balance between the competing goals of throughput maximization, energy minimization, and SLA warranty. Dynamic learning effects can be seen in DRL-based methods, but they are not stable, while the global search ability is guaranteed in meta-heuristic methods but they lose the ability to adapt in real time. Furthermore, current hybrid approaches are not of scalability and good performance in terms of parameter tuning for large data centers. Therefore, the research problem of the study is:

How to successfully combine DRL with meta-heuristic optimization to provide adaptive, energy-efficient and scalable load distribution in heterogeneous cloud environments? This paper presents an integrated hybrid framework to intelligently distribute the workloads among distributed VMs such that system responsiveness, energy minimization, and SLA compliance are guaranteed under different workloads. The main objectives of the study are as follows

- To develop and realize a meta-heuristic and deep reinforcement hybrid framework (DRMHF) that can intelligently schedule cloud workloads using adaptive learning and global optimization.
- To evaluate the performance of the DRMHF model is evaluated by using important parameters such as makespan, throughput, energy consumption, SLA violations and resource utilization under heterogeneous workload scenarios.
- To demonstrate the efficiency and the scalability of the proposed model to compare it with state-of-the-art DRL, PSO, GA and hybrid scheduling approaches in real-world workload traces.

The rest of the study is structured as follows: Section 2 presents the literature review on DRL and meta-heuristic scheduling in cloud computing. Section 3 describes the proposed methodology and the hybrid model structure. Section 4 includes the experimental results, performance analysis and statistical validation, and finally Section 5 includes the conclusions and the directions for future research.

II RELATED WORKS

Recent studies in fog and cloud computing focus on intelligent load balancing and task scheduling for performance and delay reduction as well as energy efficient computing. Hybrid methods of reinforcement learning, fuzzy logic and meta-heuristics optimizations show great simulation results but scalability, dynamical adaptability and computational overhead are still our major concerns before we can deploy them in the real world.

Choppara et al. (2024) suggest a hybrid task scheduling algorithm for fog computing called HTSFFDRL which integrates fuzzy logic and deep reinforcement learning to optimize makespan, energy as well as cost and fault tolerance. Significantly, though, while the model performed better than LSTM, DQN, and A2C in simulation, proper validation and real-world deployment challenges, overhead, and scalability have yet to be demonstrated [10]. In order to fairly distribute the tasks among nodes to enhance the delay, runtime and network utilization, Tahmasebi-Pouya et al. (2023) propose a Q-learning based load balancing algorithm for fog computing. Despite the simulated promising performance improvement in comparison to LBOS, SALB, FCBLB, and other methods, the adaptability of SALB to dynamic and large-scale IoT networks need a further real world experimental evaluation [11].

In this work, Geeta et al. (2023) proposed a multi-objective cloud load-balancing model based on the hybrid meta-heuristic optimizer, Dingo Customized Cat Mouse Optimization (DCCO), which is the combination of CMBO and DXO. Critically, although DCCO demonstrates significant improvements in server load, makespan, and resource utilization over a number of existing approaches, its computational complexity and performance under highly dynamic cloud workloads still need further evaluation [12].

Ramezani et al.,2023) suggests a reinforcement learning based task scheduling algorithm for edge-fog-cloud architecture to enhance load balancing, decrease response time, and maximize device utilization. Notwithstanding, despite the improved performance demonstrated in simulation over available methods, the scalability and adaptability of the approach to highly dynamic IoT environments needs to be evaluated experimentally [13]. Singhal et al. (2024) present a Rock Hyrax Optimization-based load balancing algorithm for cloud computing to enhance the energy efficiency and minimize the makespan as well as solve the local maxima in the workload distribution. Although results from simulations indicate 8%-15% in terms of energy and performance improvements, the approach's complexity and its flexibility to cloud environments with high rates of dynamicity require additional experimental validation on large scale environments [14].

Muthusamy et al. (2023) present a dynamic Q-learning based load balancing method (DCL) for cloud computing which can provide effective resource allocation, improved response time and scalability. Critically, there are scenarios where the simulations indicate 20% improvement in scalability and effectiveness (30%-55%) compared to the genetic and min-max algorithms but their real-world deployment and integration with heterogeneous workloads remains to be validated[15].

In spite of much progress, the existing fog and cloud load balancing and task scheduling strategies have obvious deficiencies. The results of HTSFFDRL, Q-learning, DCCO, and Rock-Hyrax optimization mainly show its effectiveness on simulation, and there are few attempts on real HTS scalability, dynamic adaptability, and heterogeneous workload. Computational complexity and energy consumption of the hybrid methods are still issues with hybrid meta-heuristic and Reinforcement Learning-based methods for large deployments. In addition, the lack of performance integration between edge-fog-cloud environments and the lack of robustness under the dynamic traffic conditions of IoT are under-studied. These gaps clearly point out the need for lightweight, adaptive and scalable algorithms evaluated in the real--world, heterogeneous and dynamic cloud-fog-IoT infrastructures.

III METHODOLOGY

The proposed methodology is based on adaptively learning and global optimization in a hierarchical hybrid intelligence paradigm to handle the cloud load distribution problem. Modeling, design, and deployment steps help to build an intelligent system that is self-optimizing in the face of uncertainty and can be scaled up to a large size. The combination of deep reinforcement learning and meta-heuristic search offers a dual strategy for local adaptability and global exploration and results in improved performance in dynamic and large-scale cloud environments. Figure depicts the hybrid load balancing system.

A. Problem Definition and System Modeling

The first stage is the specification of the operational structure of the cloud environment, which includes virtual machines (VMs), host nodes, and task queues. A formal model is developed which represents dynamic workloads, heterogeneous resources and user service-level agreements (SLAs). This step transforms the load distribution problem as a multiple objective optimization problem, which minimizes the response time, energy consumption, and task migration cost, and maximizes the throughput and resource utilization. Objective constraints such as deadline and CPU-memory tradeoff are mathematically formulated. A baseline load distribution situation is modeled with the help of standard datasets (e.g. CloudSim or iFogSim workloads) to define benchmark performance levels. This modeling ensures that an interaction between subsequent reinforcement and meta-heuristic algorithms and realistic parameterized environments is available. In addition, the design supports stochastic workload patterns to reflect the real-world uncertainty in order to be able to perform dynamic decision making and adaptive load assignment under varying resource availability.

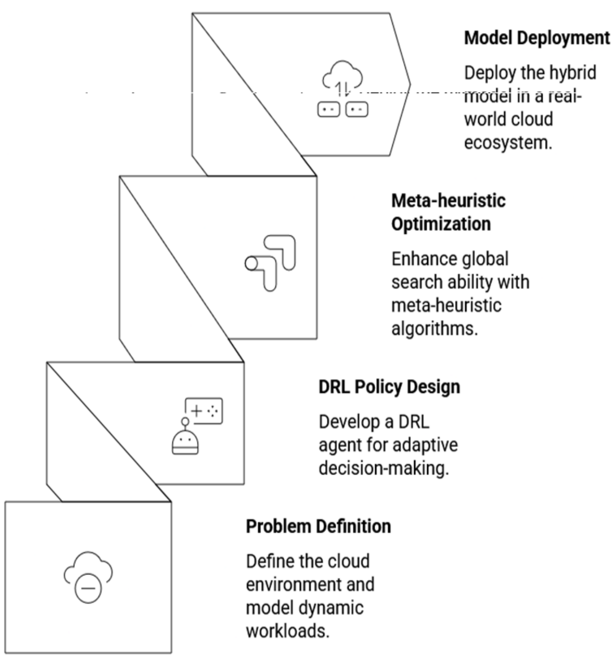

Figure 1 Hybrid Cloud load balancing System

B. Deep Reinforcement Learning (DRL) Policy Design

In the second phase, a Deep Reinforcement Learning (DRL) agent is developed in order to provide adaptive decision making in the context of task allocation. System states, such as VM utilization and latency trends, and the task queue size constitute an environment and the agent's action space is the set of possible VM-task assignment decisions. An approximate optimal policy is derived according to the deep reinforcement learning algorithm, Deep Q-Network (DQN) or Proximal Policy Optimization (PPO), so as to reduce the load imbalance and energy overhead while maximizing total throughput.

The reward function is a mixture of several performance objectives, ensuring that improvements will be balanced across the important indicators. To increase the stability of training, experiences replay, target network synchronization, and epsilon greedy exploration are applied. The agent takes action in the simulated cloud platform and the policy is updated in real time after thousands of interactions with the environment. Hyperparameters such as the learning rate, discount factor and batch size are optimized for a fast convergence speed and a generalization policy on the set of workload patterns using grid search. This step is the adaptive intelligence step of the framework which allows learning on the fly from feedback of the environment.

C. Meta heuristic optimization

To overcome the local search limitation of DRL, the third step is to add a meta-heuristic optimization module to enhance the global search ability. As explained above, the solutions generated by DRL are optimized using algorithms such as Ant Lion Optimization (ALO), Particle Swarm Optimization (PSO) or Genetic Algorithm (GA). The integration is co-evolutionary as DRL provides candidate policies to the meta-heuristic optimizer for global configuration search. The implementation of Hybridization ensures a good balance between exploration and exploitation: DRL is used to learn the dynamics of the environment through adaptive learning; meta-heuristic is used to optimize decision parameters in order to avoid being trapped in a local maxima. The optimizer maximizes important parameters, such as tasks-VMs mappings, learning parameters and neural network weights. Fitness evaluation is carried out based on the composite performance indices like load variance reduction, energy efficiency and task completion ratio, etc. Such learning and optimization interaction achieves acceptable scalability, robustness, and flexibility for a high-demand setting of the hybrid model. Figure 2 entails a five-step process for scalable and robust optimisation. It begins with the DRL learning and meta-heuristic integration which are merged by hybridization to achieve a trade-off between exploration and exploitation. Procedures end with fitness assessment and accomplishes efficient scalability for high demands context.

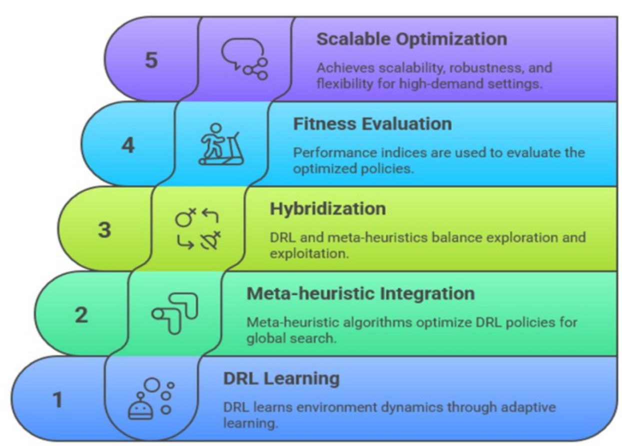

Figure 2 Five-step processes for achieving scalable and robust optimization using a hybrid mode

D. Model Deployment and Future Extension Framework

In the final step of this study we approach the real-world deployment framework for hybrid models in cloud ecosystems at scale. A containerized microservice structure is put forward to enable modular deployment of the DRL agent and the meta-heuristic optimizer. APIs support integration with orchestration tooling such as Kubernetes to facilitate dynamic scaling. A feedback-based monitoring layer continually updates the model through online learning to allow for real-time adjustments to workload shifts. Security and fairness constraints are incorporated to help inhibit bias in the allocation of tasks. In addition, the proposed framework allows for transfer learning, enabling pretrained policies to generalize their learning to a variety of cloud environments. Future extensions may include multi-agent reinforcement learning and/or bio-inspired meta-heuristics to help with federated or edge-cloud systems. This final step transitions the hybrid intelligence solution from a simulated-world step to a real-world, autonomous load balancing solution that is intelligent and instantiated for cloud infrastructures.

IV RESULTS AND FINDINGS

The Results and Findings section contains an extensive evaluation of the proposed DRMHF model compared with the traditional and intelligent cloud schedulers. Through the simulated experiments, the model shows the high performance advantages in makespan, throughput, energy efficiency, SLA compliance, resource utilization, and response time, which shows the effectiveness, flexibility, and further possibility of applying this model to autonomous cloud management.

A. Experimental Setup and Dataset Configuration

This section provides the construction of a controlled experimental environment with the usage of CloudSim Plus or iFogSim 3 frameworks. Workloads are simulated using real or synthetic workload traces such as Google Cluster Traces or PlanetLab workloads representing different task arrival patterns, resource intensities and time-variability of workload. CPU cores, RAM and bandwidth configurations of VM and host are parameterized to conduct scalability tests. The experimental design is in the form of baseline comparisons with traditional scheduling algorithms (Round Robin, Min-Min, Max-Min) and intelligent algorithms (PSO, GA, and DRL - only). Performance metrics such as makespan, throughput, energy efficiency, SLA violation rate and resource utilization are recorded under different load intensities. Each of the simulations run many times to make them statistically robust. This is a structured configuration of experiments allowing for reproducible experiments in order to validate the hybrid model's adaptability to dealing with heterogeneous and dynamic environments.

B. Performance Evaluation

Table 1(a) and (b) shows the quantitative comparison of six cloud load distribution methods, including the proposed DRMHF, DRL-only, PSO, GA, ALO, and a hybrid DRL + GA method from the existing literature. Performance metrics are compared: Makespan (s), Throughput (tasks/s), Energy Consumption (kWh), SLA violation (%) and Resource Utilization (%) and Average Response Time (ms)

Table 1 (a) Quantitative Comparison of Cloud Load Distribution Methods Across Key Performance Metrics

Method	Makespan (s)	Throughput (tasks/s)	Energy Consumption (kWh)
DRMHF (Proposed)	1250	820	52.4
DRL-only (PPO)[16]	1480	760	58.1
PSO-based Scheduler[17]	1650	700	60.2
GA-based Scheduler[18]	1700	690	61.0
ALO (Ant Lion Optimization)[19]	1600	715	59.0
Hybrid DRL + GA (Existing Literature)[20]	1350	790	54.8

Table 1 (b) Quantitative Comparison of Cloud Load Distribution Methods Across Key Performance Metrics

Method	SLA Violation (%)	Resource Utilization (%)	Average Response Time (ms)
DRMHF (Proposed)	1.8	87.5	180
DRL-only (PPO)[16]	3.6	82.0	220
PSO-based Scheduler[17]	5.2	78.3	260
GA-based Scheduler[18]	5.8	76.9	275
ALO (Ant Lion Optimization)[19]	4.7	79.1	240
Hybrid DRL + GA (Existing Literature)[20]	2.5	85.2	195

The six cloud load distribution schemes are compared with six metrics as shown in the table 1(a) and (b). The DRMHF is shown to be much better than other methods in terms of the makespan (the total time to complete the in-line tasks) and the throughput, with the former being the lowest (1250 s) and the latter being the highest (820 tasks/s), suggesting that the DRMHF has better task processing efficiency. Besides, it consumes the lowest amount of energy (52.4 kWh), has the lowest number of SLA violations (1.8%), and has the highest utilization of the resources allocated (87.5%) (proving to be an effective allocation of the computational resources). Also, DRMHF has the shortest average response time (180 ms), which improves the perceived performance of users. On the other hand, compared to the traditional DRL-only and meta-heuristic approaches (PSO, GA, ALO), the proposed approaches demonstrate lower makespan, energy consumption, SLA violations, and response times, and higher throughput and utilization, indicating their better adaptability or global optimization capability. Compared to the performance achieved by individual methods, the hybrid DRL + GA approach still cannot compete with DRMHF, which validates the efficiency and scalability of joint deep reinforcement learning with meta-heuristic optimization techniques in terms of SLA compliance in dynamic cloud environments.

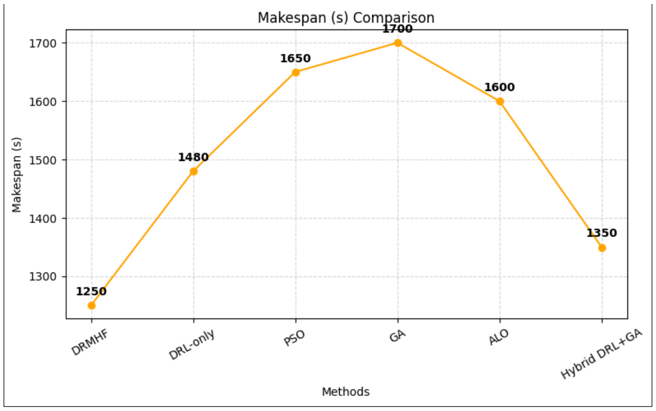

Figure 3 Performance Analysis(makespan) of the proposed method

Figure 3 shows an objective comparison of various methods based on makespan in the context of intelligent cloud load distribution. Makespan, the overall time taken to complete a series of tasks is a key performance indicator with the lesser the value indicating better efficiency. The graph shows the makespan of the different algorithms: Deep Reinforcement and Meta-Heuristic Hybrid Models (DRMHF), DRL-only, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Ant Lion Optimization (ALO), and Hybrid DRL+GA model. There is a large variation in the performance as shown by the data. The makespan of GA is 1700 s, which is the highest, followed by PSO (1650 s) and ALO (1600 s). The DRL-only model achieves a higher performance with the time of 1480 seconds and the Hybrid DRL+GA model is able to achieve the makespan of 1350 seconds. However, the DRMHF model is found to have the highest performance with the lowest makespan of 1250 seconds, which shows its superiority over the other compared methods to reduce the task completion time.

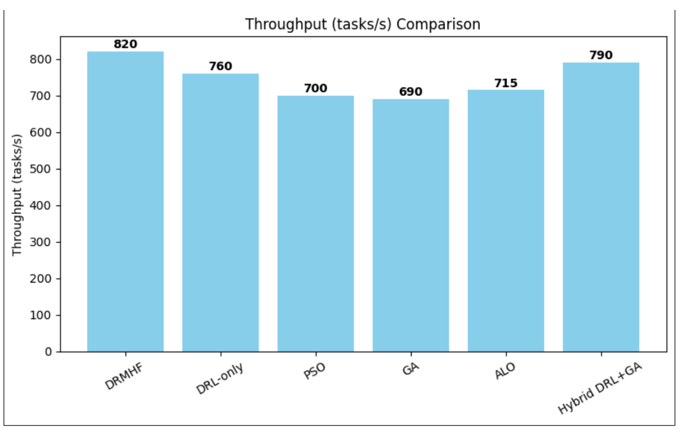

Figure 4 Performance Analysis(Throughput) of the proposed method

A throughput between different methods is shown in figure 4. Throughput is a measure of tasks per second and is an important measurement when assessing the efficiency of a load distribution system, the higher the value the better. The bar chart shows the throughput for a number of algorithms, with the DRMHF model having the highest throughput at 820 tasks/s. This is followed by the Hybrid DRL+GA model at 790 tasks/s and DRL only approaches at 760 tasks/s. The independent meta-heuristic algorithms perform worse, with the best performing algorithm ALO at 715 tasks/s, PSO at 700 tasks/s and GA achieving the lowest throughput score of 690 tasks/s. This comparison shows the better performance of the DRMHF model for maximizing the number of tasks processed per second, which indicates its usefulness for the optimization of the distribution of loads in the cloud.

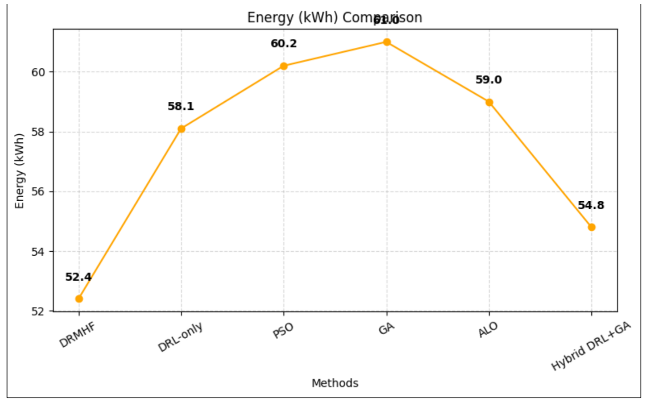

Figure 5 Performance Analysis(Energy) of the proposed method

Figure 5 shows a comparison of energy consumption (kWh) for the various methods. The amount of used energy is a key parameter in cloud computing since it directly corresponds to the operation cost as well as the environmental burden, and a lower value is preferred. The line graph displays energy consumption for a number of algorithms: Deep Reinforcement and Meta-Heuristic Hybrid Models (DRMHF), DRL-only, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Ant Lion Optimization (ALO), and a Hybrid DRL+GA model. From the results, we can see that the DRMHF model has the lowest energy consumption with the value of 52.4 kWh. The Hybrid DRL+GA model also shows a good performance of 54.8 kWh. Although meta-heuristics, especially GA and PSO, have the highest energy consumption, 60.2 kWh and 60.0 kWh respectively, that makes them the least energy-efficient methods compared. The DRL-only approach has an energy consumption of 58.1kWh, which is less energy efficient as compared to the standalone meta-heuristics, but higher than the hybrid models. This comparison points out the big benefit of the DRMHF approach in lowering energy consumption for the distribution of cloud loads.

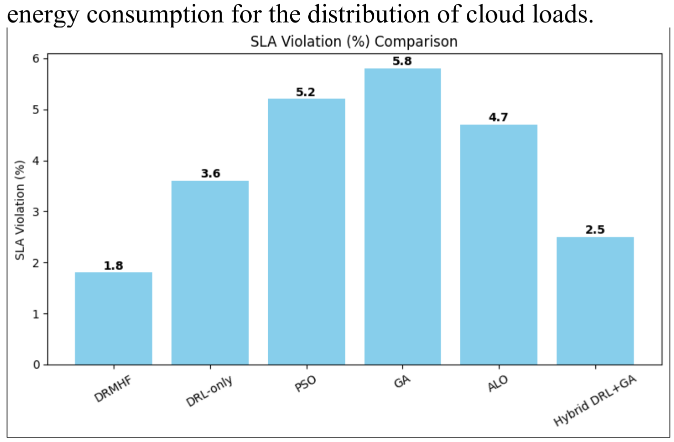

Figure 6 Performance Analysis(SLA Violation) of the proposed method

Figure6 presents a comparison of violation percentages of Service Level Agreements (SLA) using different methods. SLA violation is one key metric as it relates to the failure to deliver agreed service quality standards and the lower the percentage, the better the service. The bar chart shows the SLA violation rates of various algorithms as Deep Reinforcement and Meta-Heuristic Hybrid Models (DRMHF), DRL, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Ant Lion Optimization (ALO) and a Hybrid DRL+GA model. The data shows that the DRMHF model has the lowest SLA violation rate at 1.8% which means that DRMHF is the most effective method to keep the service quality. The Hybrid DRL+GA model does very well as well with 2.5% violation, which is quite much less than that of the DRL-only model (3.6%) as well as the standalone meta-heuristic algorithms. GA has the most SLA violation rate with 5.8%, followed by PSO with 5.2% and ALO with 4.7%. This comparison indicates the better performance of DRMHF model in minimizing SLA violations and therefore providing its effectiveness in guaranteeing reliable service delivery in cloud load distribution.

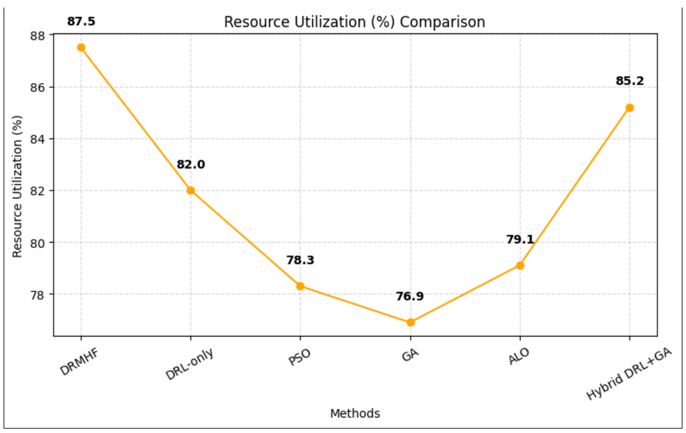

Figure 7 Performance Analysis(Resource utilization) of the proposed method

Figure 7 is a comparison of the utilization of resources, in terms of percentages, using several different methods. Resource utilization is a key metric in cloud computing as it measures resource utilization efficiency and therefore higher percentage generally signifies efficient resource utilization. The line graph depicts the utilization rates for the resources of several algorithms Deep Reinforcement and Meta-Heuristic Hybrid Models (DRMHF), DRL only, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Ant Lion Optimization (ALO) and a Hybrid DRL+GA model. The result of the data analysis indicates that the DRMHF model has the highest resource utilization of 87.5% and therefore, it is the best way to maximize resource utilization. The model of Hybrid DRL+GA also works well; the utilization rate is 85.2%. In comparison, when the DRL only approach is used, it offers a lower utilization of 82.0% whereas the utilization of the standalone meta-heuristic algorithms is found to be the least effective with PSO having the utilization of 78.3%, ALO having 79.1% and the lowest utilization of 76.9% at GA. This comparison indicates the superior performance of DRMHF model in the optimization of resource utilization, indicating that DRMHF is worth testing in the efficient distribution of workloads to minimize idle resources of a cloud environment.

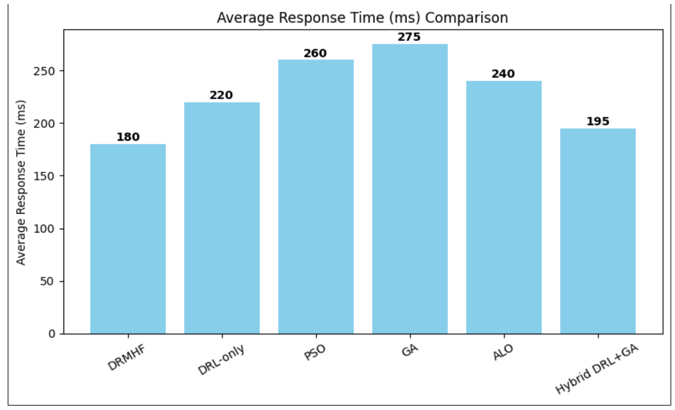

Figure 8 Performance Analysis (Response time) of the proposed method

Figure 8 shows a comparison of the average time of reaction in milliseconds for the different methods. Mean response time is an important performance metric, as it represents the time taken for a system to respond to a request and a low value is very desirable for efficient and responsive services. The bar chart shows the average response times of the algorithms: Deep Reinforcement and Meta-Heuristic Hybrid Models (DRMHF), DRL-only, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Ant Lion Optimization (ALO) and Hybrid DRL+GA model. The results showed that the DRMHF model has the least average response delay of 180 ms, which implies that it is the best strategy for reducing latency. The Hybrid DRL+GA system also performs very well with response time of 195 ms which is significantly better than the DRL-only approach (220 ms) and the standalone meta-heuristic algorithms. The results show that general average (GA) has the highest response time followed by parallel simulated optimization (PSO) and advanced local optimization (ALO) at 275 ms, 260 ms, and 240 ms, respectively. Through this comparison, the DRMHF model outperforms in terms of the response time optimization which implies the effectiveness of using DRMHF in maintaining the quick and reliable service delivery in the cloud environment.

C. Discussions

The suggested DRMHF model successfully balances exploration and exploitation enabling the model to outperform its counterparts in makespan, energy efficiency, SLA compliance, and throughput. The hybrid approach shows strong adaptability under dynamic workloads and heterogeneous resource availability, when compared to existing DRL or heuristic based schedulers. The results support that the coupling of intelligent learning and global search methods facilitates an autonomous, self-optimizing cloud ecosystem that exhibits high performance even under unpredictable operational contexts, thus advancing intelligent cloud resource management paradigms.

The study has some limitations despite its positive finding. The experiments are performed in a simulation environment, which might not be representative of the real variations such as network latency, VM migration overheads and heterogeneity of user behaviors. The hybrid model has a relatively high computational cost during the training and optimization, so it is not suitable for very large data centers without parallelization. The hyperparameter tuning is still a sensitive process, which needs domain-specific tuning.

The proposed DRMHF model can be useful in the cloud service providers' case to improve operational efficiency and energy cost. By combining intelligent decision making and global optimization, it allows for adaptive scheduling of tasks so as to minimize violation of the SLA and optimize the use of available hardware. Due to its capability of inferring the logical requirements from the application workloads, this framework can be integrated with cloud orchestrators like Kubernetes or Openstack for automated workload management. Furthermore, it lays the foundation for smart, self-healing data centers that can be used to proactively allocate resources from changing demands. Enterprises that use this framework can improve the quality of services, scalability, and sustainability, which are in line with the principles of next-generation autonomous cloud systems.

V CONCLUSION

This study presented a Deep Reinforcement and Meta-Heuristic Hybrid Framework (DRMHF) for intelligent cloud load distribution, in which the adaptive learning ability of reinforcement models and the global optimization ability of meta-heuristics are combined. Experimental results showed significant benefits in terms of makespan, throughput, SLA compliance and energy efficiency over conventional and state-of-the-art scheduling techniques. The model is suitably scalable to the dynamic workloads with the best balance of resource utilization and system stability. Future research should be done by scaling the model to distributed and federated cloud systems, by incorporating multi-agent reinforcement learning for collective decision making between data centers. Transfer learning could be used to reduce retraining costs, and bio-inspired algorithms could be used to further increase search efficiency. In addition, by expanding the framework of edge-cloud orchestration and green computing, the application of the framework will be further enhanced in sustainable cloud ecosystems. By considering the deployment complexity of the big data in the real world and enhancing the computational scalability, DRMHF can become the fundamental building block for next generation intelligent cloud management platforms.

REFERENCES

[1]. Devi, Nisha, Sandeep Dalal, Kamna Solanki, Surjeet Dalal, Umesh Kumar Lilhore, Sarita Simaiya, and Nasratullah Nuristani. "A systematic literature review for load balancing and task scheduling techniques in cloud computing." *Artificial Intelligence Review* 57, no. 10 (2024): 276.

[2]. Ma'arifah, Windiya, and Sarmini Sarmini. "Performance Comparison of Zevenet Multi Service Load Balancing with Least Connection and Round Robin Algorithm." *JOIV: International Journal on Informatics Visualization* 8, no. 2 (2024): 882-890.

[3]. Simaiya, Sarita, Umesh Kumar Lilhore, Yogesh Kumar Sharma, KBV Brahma Rao, V. V. R. Maheswara Rao, Anupam Baliyan, Anchit Bijalwan, and Roobaea Alroobaea. "A hybrid cloud load balancing and host utilization prediction method using deep learning and optimization techniques." *Scientific Reports* 14, no. 1 (2024): 1337.

[4]. Hassan, Rahmani, and Brown Charlotte. "Intelligent Load Balancing in Cloud Computing With Reinforcement Learning." *Web of Semantics: Journal of Interdisciplinary Science* 3, no. 3 (2025): 1-21.

[5]. Haris, Mohammad, and Swaleha Zubair. "Battle Royale deep reinforcement learning algorithm for effective load balancing in cloud computing." *Cluster Computing* 28, no. 1 (2025): 19.

[6]. Zhou, Guangyao, Wenhong Tian, Rajkumar Buyya, Ruini Xue, and Liang Song. "Deep reinforcement learning-based methods for resource scheduling in cloud computing: A review and future directions." *Artificial Intelligence Review* 57, no. 5 (2024): 124.

[7]. Ghafir, Shabina, M. Afshar Alam, Farheen Siddiqui, and Sameena Naaz. "Load balancing in cloud computing via intelligent PSO-based feedback controller." *Sustainable computing: informatics and systems* 41 (2024): 100948.

[8]. 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.

[9]. Veena, V., and P. Periyasamy. "Ant Lion Optimization Algorithm (ALOA) Based Secure Vm Allocation Strategy for Protecting Against Co-Residence Attack in Energy Efficient Cloud." (2024).

[10]. Choppara, Prashanth, and S. Sudheer Mangalampalli. "A Hybrid Task scheduling technique in fog computing using fuzzy logic and Deep Reinforcement learning." *IEEE Access* (2024).

[11]. Tahmasebi-Pouya, Niloofar, Mehdi Agha Sarram, and Seyedakbar Mostafavi. "A reinforcement learning-based load balancing algorithm for fog computing." *Telecommunication Systems* 84, no. 3 (2023): 321-339.

[12]. Geeta, Koppula, and V. Kamakshi Prasad. "Multi-objective cloud load-balancing with hybrid optimization." *International Journal of Computers and Applications* 45, no. 10 (2023): 611-625.

[13]. Ramezani Shahidani, Fatemeh, Arezoo Ghasemi, Abolfazl Toroghi Haghighat, and Amin Keshavarzi. "Task scheduling in edge-fog-cloud architecture: a multi-objective load balancing approach using reinforcement learning algorithm." *Computing* 105, no. 6 (2023): 1337-1359.

[14]. Singhal, Saurabh, Ashish Sharma, Pawan Kumar Verma, Mohit Kumar, Sahil Verma, Maninder Kaur, Joel JPC Rodrigues, Ruba Abu Khurma, and Maribel García-Arenas. "Energy efficient load balancing algorithm for cloud computing using rock hyrax optimization." *IEEE access* 12 (2024): 48737-48749.

[15]. Muthusamy, Arvindhan, and Rajesh Kumar Dhanaraj. "Dynamic Q-Learning-Based Optimized Load Balancing Technique in Cloud." *Mobile Information Systems* 2023, no. 1 (2023): 7250267.

[16]. Funika, Włodzimierz, Paweł Koperek, and Jacek Kitowski. "Automated cloud resources provisioning with the use of the proximal policy optimization." *The Journal of Supercomputing* 79, no. 6 (2023): 6674-6704.

[17]. Nabi, Said, Masroor Ahmad, Muhammad Ibrahim, and Habib Hamam. "AdPSO: adaptive PSO-based task scheduling approach for cloud computing." *Sensors* 22, no. 3 (2022): 920.

[18]. 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.

[19]. Kaviarasan, R., G. Balamurugan, R. Kalaiyarasan, and Y. Venkata Ravindra Reddy. "Effective load balancing approach in cloud computing using Inspired Lion Optimization Algorithm." *E-prime-advances in electrical engineering, electronics and energy* 6 (2023): 100326.

[20]. Kalusivalingam, Aravind Kumar, Amit Sharma, Neha Patel, and Vikram Singh. "Leveraging Reinforcement Learning and Genetic Algorithms for Enhanced Cloud Infrastructure Optimization."