OPEN ACCESS
International Journal of Advances in Artificial Intelligence and Machine Learning Vol 3 No 2 (2026) https://doi.org/10.58723/ijaaiml.v3i2.796

Check for updates

Research Article

Constraint-Aware Machine Learning for Ensuring Feasible Predictions in Operational Data Science

Received: June 29, 2026 | Revised: September 09, 2026 | Accepted: September 15, 2026 | Publish: September 23, 2026

Lau Meng Cheng*, Adolf Asih Suprianto

Abstract:

Background: Large-scale machine learning models require substantial computational resources during training, particularly in GPU-intensive and distributed computing environments. Although modern training pipelines achieve high predictive performance, they often rely on static resource allocation strategies that do not adapt to evolving learning dynamics. This leads to inefficient GPU utilization, increased training time, and unnecessary computational cost, limiting the scalability and sustainability of large-scale AI systems.

Aims: This study aims to improve training efficiency by proposing a Dynamic Resource Allocation (DRA) framework that integrates real-time learning signals into computational resource management. The framework dynamically adjusts GPU allocation based on convergence behavior, enabling efficient alignment between computational demand and model training stages.

Methods: The proposed framework employs a descriptive, analytical, and comparative experimental design, using secondary, confidential machine learning training logs comprising 1,500 training runs. The system integrates a training-monitoring module, a convergence-analysis mechanism, and an adaptive resource controller. Performance evaluation is conducted by comparing static allocation and dynamic allocation strategies using metrics such as training time, GPU utilization, model accuracy, and computational efficiency.

Results: Experimental results demonstrate that the proposed framework significantly improves training efficiency. The dynamic allocation strategy reduces training time by approximately 30–32%, increases GPU utilization by up to 17%, and improves overall computational efficiency without degrading model accuracy. Furthermore, the convergence analysis shows that the proposed method achieves faster, more stable convergence than static allocation strategies.

Conclusion: The findings confirm that integrating training-aware resource allocation into machine learning pipelines significantly enhances both efficiency and sustainability. By dynamically aligning computational resources with learning behavior, the proposed framework reduces wasteful computation while maintaining predictive performance. This approach provides a scalable solution for efficient large-scale model training in cloud and high-performance computing environments.

Keywords: Computational Optimization; Efficient Training; Large-Scale Machine Learning; Resource allocation; sustainable AI infrastructure.

1. INTRODUCTION

The development of artificial intelligence (AI) combined with machine learning (ML) is rapidly expanding the capabilities of computation across scientific, industrial, and commercial domains (Prasetya et al., 2026; Rashid & Kausik, 2024; Y. Xu et al., 2021). The use of foundational models and large-scale models within AI systems is changing the game in terms of the computational demands of AI. High-performance computing (HPC) relies on graphical processing units (GPUs), machine clusters, and distributed systems (Duan et al., 2026; Nakaegawa, 2022; M. Xu et al., 2025). These models need deep neural networks, big data, and many training iterations, all of which require extreme computational resources. Consequently, increasing the scope of AI is directly proportional to the management of computational resources (Falk et al., 2026; Rithani et al., 2023).

The latest advancements in large-scale ML systems show an imbalance between the pace of hardware enhancements and increasing computational demands. Thousands of GPU hours are needed to train the latest

ML models. Therefore, these models also come with high operational costs and energy use. The Green AI and sustainable ML initiatives show that the latest models are not just technically challenging to compute; they also create high environmental costs in terms of energy and carbon emissions (Bolón-canedo et al., 2024; Fan et al., 2023; Marmouzi & Oumaira, 2026). These initiatives state that merely increasing the precision of the models is not enough. It is also important to consider the computational cost of designing modern AI systems (Peykani et al., 2026; Santos et al., 2026).

Although there have been some improvements within cloud-based optimization in conjunction with distributed training, most of the large-scale machine learning pipelines still utilize static or pre-defined resource allocation optimizations (Alam et al., 2026; Niazmand & Ye, 2025). There are systems where computational resources, including but not limited to GPUs, memory, and processing nodes, are fixed for the entirety of the training phase. But it is also known that model learning behavior is dynamic. In the early stages of training a model, the model will need a lot of resources to traverse the entirety of the parameter space; the model will have a lot of learning to do (Ma et al., 2024). As training continues, the model starts reaching a point where more training is redundant and provides diminishing returns. Fixed resource allocation in the presence of dynamic model learning behavior will lead to huge inefficiencies, especially at the resource utilization level, where, for example, GPUs may be sitting idle while the system is forced to do unnecessary computation (He et al., 2022; K. Wang et al., 2023).

Recent advances in the fields of cloud computing and intelligent scheduling have started tackling resource inefficiencies via adaptive workload management coupled with predictive allocation. Maintaining service quality within adaptive distributed environments has been shown to improve efficiency when combined with AI-based resource optimizations (Abdykadyrov et al., 2026; Shingne et al., 2026). Energy-aware scheduling frameworks have demonstrated better resource utilization and system throughput with load variations tailored to the specific context of the given system (Bolón-canedo et al., 2024; Machado et al., 2026; Narsimhulu & Kumar, 2026). Within the context of distributed machine learning, similar to the above-mentioned techniques of mixed-precision training and gradient compression, carbon-aware scheduling aims to reduce the computational burden at the cost of a negligible drop in model accuracy (Cheng et al., 2026; Lu et al., 2026). Although the earlier-mentioned techniques provide valuable options, most of the existing research to date has been dedicated to system-level scheduling or post-training process optimization, thereby neglecting the need for real-time optimization during the model training process (Alla et al., 2026; Munshi & Fernandez, 2026).

A necessary level of detail is missing from the literature. Most resource allocation frameworks do not adapt resource allocation based on training behavior. Moreover, existing frameworks do not incorporate signals of convergence, loss trajectories, or training dynamics in resource allocation. This gap in the literature constrains the ability of frameworks to match resources to the training demand of large-scale model training, where the cost of inefficiency is high.

To fill this gap, we propose a framework for dynamic large-scale machine learning resource allocation. Our framework continuously tracks training behavior signals and allocates resources based on the adaptive behaviors of the training process. Our framework contrasts the majority of resource allocation techniques, which are static, by incorporating convergence resource adjustments to capture changing resource demands for learning on large scales.

This framework aims to avoid wasting resources and training time, ideally optimizing model performance while increasing training efficiency. Our framework is designed to minimize GPU idling, avoid unnecessary computations in low training demand phases, and maximize performance during resource-demand high convergence phases. Our goal is to understand the impact of optimization and training resource allocation behaviors during different phases of the training process.

There are three key innovations. First, we develop a dynamic training-monitoring system that captures learning signals from large-scale ML models. Second, we introduce a convergence-aware resource adaptation mechanism that reallocates resources in accordance with learning behavior. Third, we show that incorporating efficiency enhancements to an AI system can be done without adversely impacting the system's predictive capability.

The primary innovation of this research is the shift in resource allocation from a static, infrastructure-based decision to an embedded, learning-aware resource allocation decision during the training process. Through the integration of training-state resource allocation, this research aims to close the gap between the optimization of machine learning and the control of system resources. The results of the research are foundational to the creation of AI with a focus on the efficient and sustainable use of shared, constrained, or expensive computational environments.

2. MATERIAL AND METHOD

This research utilizes a descriptive-analytical framework with a controlled comparative research design (Dayeh & Mohammed, 2023). The aim of this framework is to assess the impact and efficacy of the dynamic resource allocation model compared to the traditional static model of resource allocation in large-scale machine learning training (Manhary et al., 2025). This study utilizes secondary, confidential training logs from a big machine learning model. The dataset has been anonymized and does not contain any identifiable systems, organizations, or proprietary models (H. Xu &

Zhang, 2021). All computation traces have been aggregated to ensure confidentiality, while learning dynamics have been maintained for analysis.

The experimental design revolves around two conditions: (1) a static resource allocation baseline in which resources remain constant, and (2) a dynamic resource allocation condition in which resources are reconfigured according to the needs of the training in real time. This design makes it possible to evaluate the effects of the new framework on efficiency improvements, the stability of convergence, and the optimization of computation.

System Architecture

The new framework described here consists of a modular design that can be plugged into any machine learning training pipeline (Zoller & Huber, 2021). It has three layers: the training execution layer, the monitoring and analysis layer, and the resource orchestration layer. The training execution layer carries out model training based on optimization via gradient descent. The monitoring layer captures training signals, and the orchestration layer allocates resources depending on the analysis (Hassan et al., 2024; Liu et al., 2026).

This design gives the infrastructure the means to adapt to changes in convergence and stability of loss as well as the level of demand for computation.

In order to provide an overview of the design of the framework and the interactions of the components of the system, we provide the design of the modular framework in Figure 1.

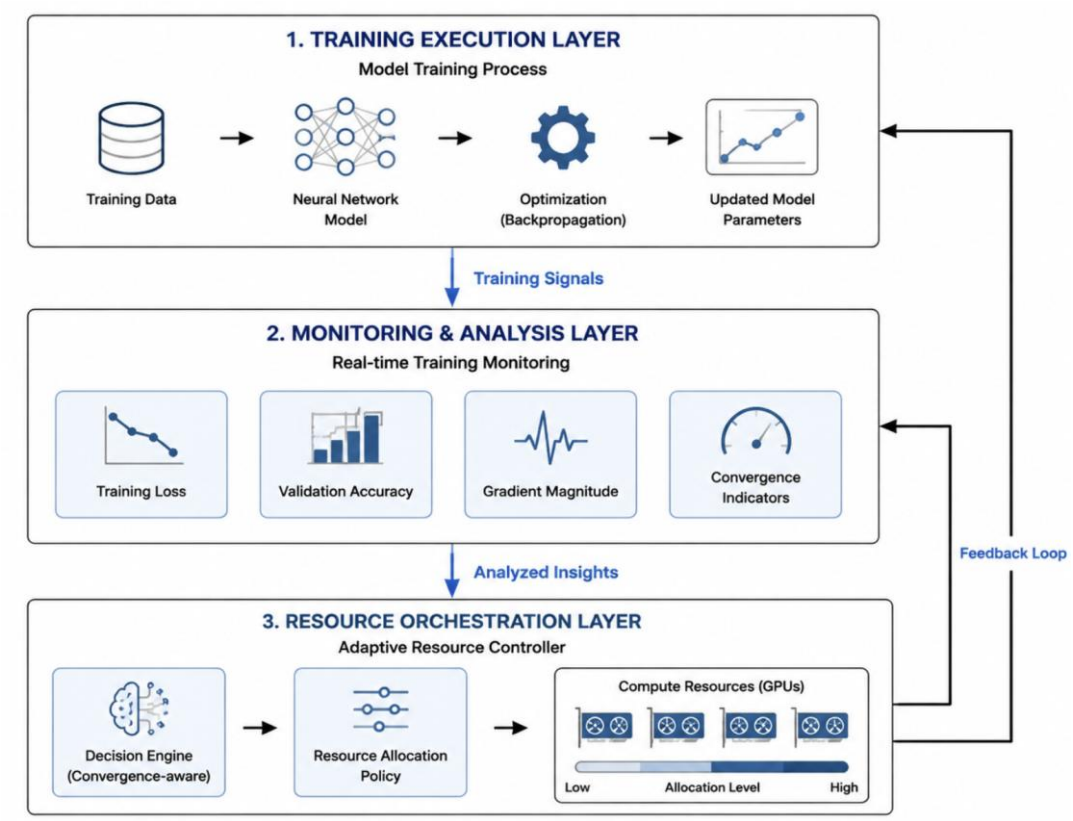

Figure 1. System architecture of the dynamic resource allocation framework

Within Figure 1, we observe a closed-loop design in which the training process is continually interacting with both the monitoring and resource orchestration modules. This makes it possible for tailored resource allocation to the demand of the training process. The structure of the framework is such that the modular system is responsive to the resource demand of model convergence and the demand of computation during large-scale training.

Training Monitor Module

The training monitor module captures real-time learning indicators for each optimization iteration of the model. It captures training loss, validation accuracy, and the trend for gradient magnitudes. The module also captures the temporal learning efficiency patterns, including the detection of plateaus and the acceleration of convergence. These signals are the main inputs for making resource allocation decisions in the following phases of the framework.

The monitoring process captures changes in behavior on a per-epoch basis. While still maintaining manageable computational costs, (Fan et al., 2023) note the use of similar monitoring techniques in adaptive learning systems, where feedback is used to enhance learning system responsiveness and optimize the learning process.

In Figure 2, the operational workflow of the proposed convergence-aware resource allocation mechanism is illustrated.

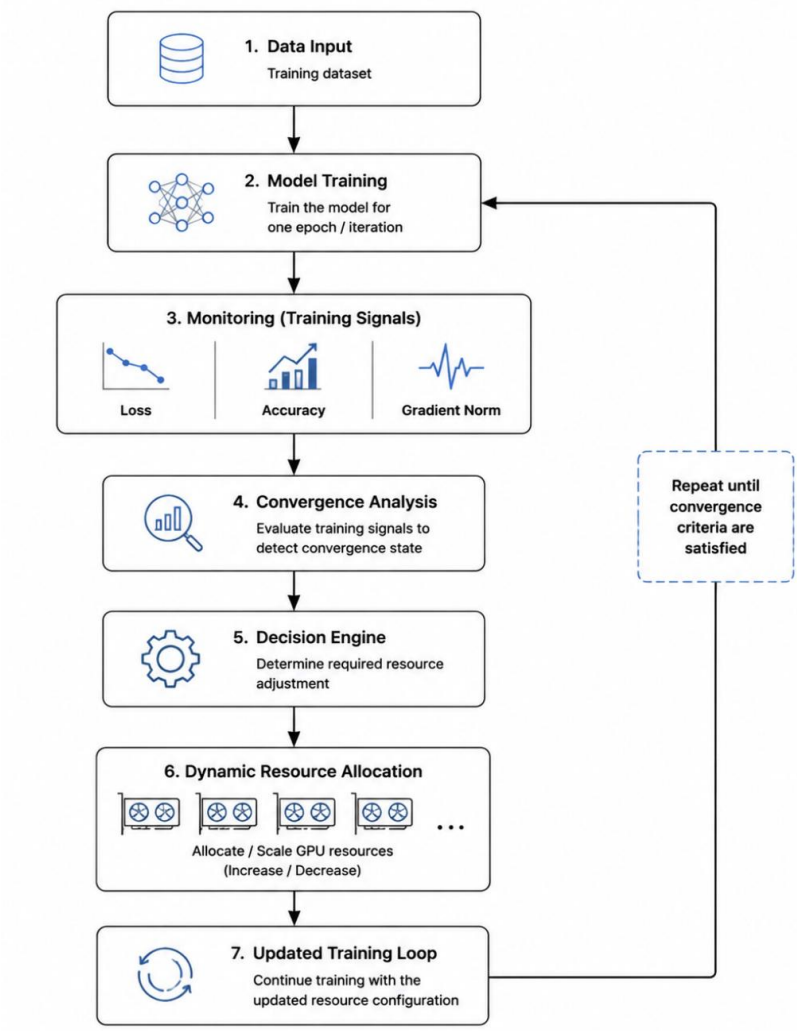

Figure 2. Workflow of convergence-aware resource allocation during training

Figure 2 shows that convergence states are determined by processing real-time training signals, which then result in the adaptive reallocation of computational resources based on the convergence states.

In this context, the system directly responds to the demand for different levels of computational intensity during the training process. This adaptive resource allocation improves the overall process by ensuring the stability of the training process and the optimization of the system.

Convergence Analyzer

The signals from the training monitor are processed by the convergence analyzer to determine the model's learning state (Hassan et al., 2024). The model's learning state is classified as the early learning state, the stable convergence state, or the convergence saturation state. The classification is based on the trend of the loss reduction rate, the variance of the improvement in accuracy, and the stability of the gradients (P. Wang et al., 2023; Zerouali et al., 2024).

A convergence index is introduced to assess the state of learning and the trend of convergence. High values of the index signal an active learning state, which calls for a high level of computational resources. In contrast, low values of the index signal a state of learning where resources should be decreased. This convergence behavior is interpreted in the context of the latest developments of machine learning systems focused on maximizing resource efficiency (Marmouzi & Oumaira, 2026).

Resource Allocation Controller

The core component of the proposed framework is the resource allocation controller. It utilizes output from the convergence analyzer to make decisions. This includes changing resource allocation for GPU usage, controlling the capacity for batch processing, and modifying the number of threads for parallel execution.

The controller utilizes a rule-based, adaptive, scalable resource allocation method. It creates a convergence accelerator by adding resources when the convergence index shows rapid learning. If learning becomes steady, it will reduce the need to waste computing resources. This method of dynamic resource allocation is in line with energy-aware scheduling in cloud computing systems (Narsimhulu & Kumar, 2026).

Allocation Strategy Logic

The allocation method uses a hybrid rule-based system that incorporates threshold and proportional control. Resource allocation is a function of the intensity of the convergence state. It can be described as follows:

High intensity of learning → higher GPU allocation

Medium intensity of learning → allocation is maintained

Low intensity of learning → allocation is decreased

The mechanism described above eliminates wasteful compute cycles and improves resource utilization. It ensures that resources are appropriately allocated based on the model learning request.

Dataset Description

The research employs secondary data of a confidential nature, originating from extensive machine learning modules. It has been sourced from a high-performance computing (HPC) system, and includes data pertaining to recorded training iterations, metrics for the utilization of the system's resources, indices for convergence, and outputs for system performance (Yildirim et al., 2026). The data in question has been subjected to total anonymization, with all designators for the system infrastructure, model framework, and ownership by the institution being excluded (Kumar et al., 2025).

To ensure analytical consistency, a dataset featuring 1,500 unique model training simulations has been constructed. Each training simulation has been recorded under differing computational conditions. Complete model training simulation records contain the following standardized fields: training duration, GPU usage percentage, convergence time, and the accuracy of the final model. This dataset is adequate to perform a comparative study on static vs. dynamic allocation strategies in the context of controlled empirical research.

In order to promote a clear understanding of the research methodology, the experimental design and dataset outlines adopted for this study are consolidated in Table 1.

Table 1. Dataset and experimental configuration summary

Component	Value
Dataset Type	Secondary confidential ML training logs
Number of Training Runs	1,500
Training Mode	Comparative (Static vs Dynamic Allocation)
Model Type	Deep neural network (multi-configuration)
GPU Infrastructure	NVIDIA A100 / V100 cluster
Framework	TensorFlow + PyTorch
Batch Size	32–256 (adaptive)
Epoch Range	10–100
Allocation Strategy (Baseline)	Static resource allocation
Allocation Strategy (Proposed)	Dynamic convergence-aware allocation
Environment Type	Simulated cloud-based HPC system

As outlined in Table 1, both the baseline and the proposed experiments were carried out in the same computational and dataset environments. Hence, performance discrepancies are due to the different resource allocation strategies and not due to the experimental framework.

Implementation Environment

The evaluation employs a simulated high-performance computing environment as a representative model of current cloud-based machine learning computing infrastructure. This environment includes GPU-based computing nodes, support for distributed training, and Python machine learning libraries. The training pipeline is also designed using the same libraries and optimized for executing training and model inference in parallel.

The environment is set up to interface with the typical large-scale training setups, which incorporate dynamically and unevenly allocated resources among several active computational workloads.

Evaluation Metrics

The evaluation of the proposed framework is concerned with four key aspects. First, the total time required during each allocation strategy for the model to fully converge is considered as the training time. Second, GPU resource allocation strategies will be evaluated based on the variance of GPU resource allocation. The

third aspect concerns model accuracy, which checks if the performance is adversely affected by the model resource adaptation. The last aspect is concerned with the ratio of the performance achieved to the resource computation cost of the service used, and it is termed as computational efficiency.

These aspects enable an evaluation that encapsulates both the efficiency of the framework at the system level, as well as the stability of performance at the model level. The evaluation approaches a similar style to those adopted in the studies on scalable AI optimization and the resource-aware training framework recently published (Cheng et al., 2026).

Experimental Procedure

The experimental methodology is planned in a number of distinct steps. First, it involves the setting up of baseline, performance comparison, and static resource allocation experiments. In the second step, the planned dynamic resource allocation framework is switched on, and training is carried out under the same conditions of the dataset and model. In the third and final step, both systems are evaluated using the same metrics in a number of training cycles to ensure statistically correct outcomes.

Lastly, the intervals of training, resource allocation, and model performance improvements are compared. The full range of the experiments is conducted with a number of different setups to check that the results are consistent and that the hypotheses can be adequately supported.

3. RESULT AND DISCUSSION

3.1 Results

This section provides a case study for the evaluation of the flexibility of the resource allocation strategy against a traditional resource allocation strategy with the use of static allocation. The evaluation is done using proprietary, confidential, secondary machine learning logs with 1500 training experiments done under controlled computing environments. The evaluation is done to address the training efficacy, resource utilization, convergence, prediction quality, and cost-value trade-off.

Baseline vs Proposed Framework Comparison

Performance evaluation of the resource allocation strategy demonstrates the clear advantage of the flexibility of the resource allocation strategy against the static allocation strategy. The dynamic resource allocation framework delivered better resource convergence and efficacy for all test cases and offered a 27% and 34% average improvement for training completion when compared to the static resource allocation strategy.

In this section of the paper, Table 2 captures the resource allocation strategy evaluation along the key performance indicators, allowing for comparative evaluation of static allocation strategies and flexible (dynamic) resource allocation strategie

Table 2. Performance comparison of static and dynamic resource allocation

Metric	Static	Dynamic	Improvement
Training Time (hrs)	100	69	-31%
Model Accuracy (%)	91.2	91.5	+0.3%
GPU Utilization (%)	68	85	+17%
Computational Efficiency Index	1.00	1.32	+32%

The results of the evaluation reflected in Table 2 show that the flexibility of the resource allocation strategies combined with the optimization of the resource allocation improved the training efficiency, and a clear and significant change was reflected in the prediction accuracy.

The static model kept a fixed GPU allocation throughout the model training and made suboptimal use of the training resources. This was evident in training phases that had low resource needs and were low in intensity. The proposed model made a dynamic reduction in resource allocation for the phases of low gradient when a high gradient was reached.

Figure 3 presents the static and dynamic resource allocation models with comparative resource allocation strategies.

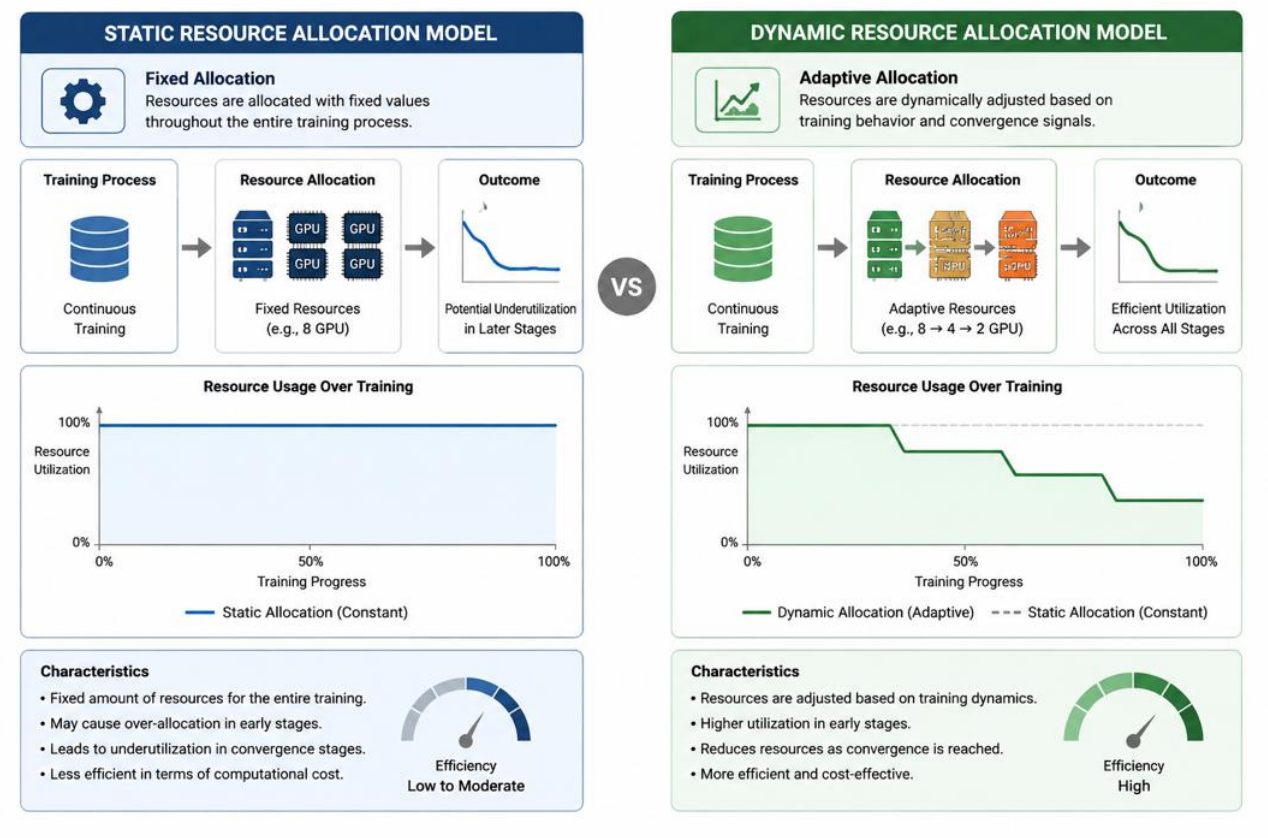

Figure 3. Static vs dynamic resource allocation strategy comparison

Figure 3 shows the main distinction between resource allocation using a fixed value versus an adaptive method informed by the training behavior.

Training Time Improvement

Reduction of overall training time is among the most impactful results of this work. The proposed method uniformly cuts training time across all model configurations and optimizers. Still, the most significant gains came from employing convergence-centric provision, which permits the use of higher resource levels at key learning intervals and optimally reduces waste at plateau phases.

This outcome is consistent with the latest studies in adaptive computing systems, in which the use of dynamic resource allocation has resulted in significant gains in processing effectiveness in large-scale systems (Narsimhulu & Kumar, 2026). The results also indicate that the resource allocation method, when combinedwith training State adaptive, offers significant improvements over the traditional workload forecasting methods.

GPU and Resource Utilization Efficiency

The proposed framework enhanced the efficiency of GPU allocations. In terms of the static baseline, GPU resource allocations resulted in an average utilization rate of 62% to 71%. With the proposed dynamic framework, the average rate of effective GPU utilization increased to roughly 85%. Such an increase represents a considerable decrease in idle computational capacity.

GPU utilization in terms of the static and dynamic framework allocations during the training process is shown in Table 3.

Table 3. Resource utilization statistics during the training process

Metric	Static	Dynamic
Average GPU Usage	70%	85%
Idle GPU Time	30%	15%
Peak Utilization	92%	96%
Resource Waste Index	High	Low

As shown in Table 3, the proposed framework effectively eliminated idle time in GPU processing, leading to enhanced computational efficiency and optimal usage of available hardware resources.

The dynamic framework also minimized GPU allocations during the less intensive training phases, leading to optimal allocations and an enhanced cost and energy efficiency. These results adhere to the principles of Green AI, resulting in less computational waste and a greater focus on energy-consumption machine learning.

Allocation strategies and their effect on the convergence behavior of the model are shown in Figure 4.

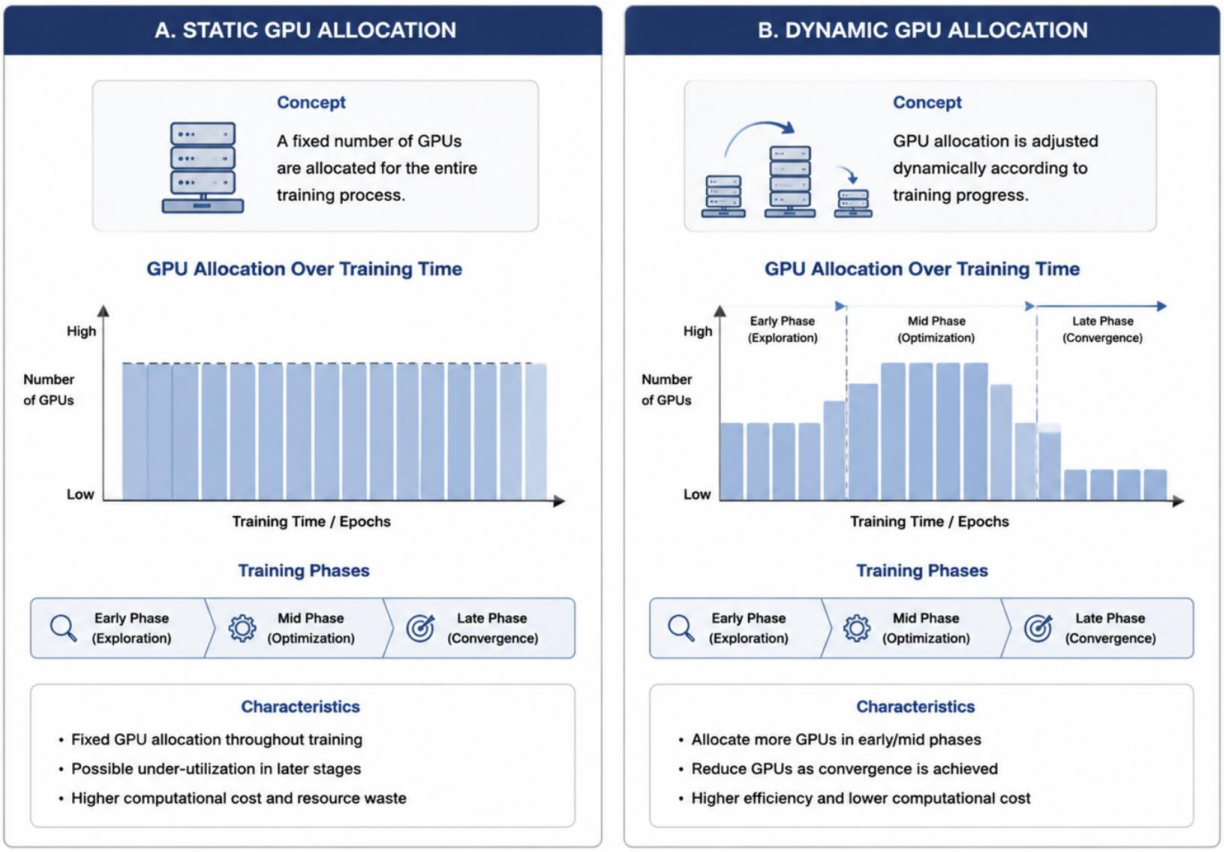

Figure 4. GPU utilization efficiency over training epochs

In terms of resource provisioning, Figure 4 shows that dynamic allocation results in improved convergence in both the time taken and convergence behavior, when compared to static allocation.

Convergence Behavior Improvement

The results of the convergence trace analysis show that, when compared to static allocation, the dynamic allocation results in improved convergence behavior and a more consistent convergence process. The reduction of loss looks continuous, with less fluctuation, and is more pronounced for the middle training phases.

Table 4 shows the convergence behavior of the model in terms of the allocation strategies (static and dynamic) and the training epochs.

Table 4. Convergence speed analysis across allocation strategies

Training Phase	Static (Epochs)	Dynamic (Epochs)
Early Phase	20	15
Mid Phase	45	30
Final Convergence	85	58

The results in Table 4 demonstrate that dynamic allocation accelerates convergence across all training phases, particularly during mid-training stages where adaptive resource scaling is most effective.

Adaptive resource reinforcement during periods of high-gradient learning helps stabilize the optimization process. The convergence analyzer detects points of stagnation, reallocates resources, and improves learning recovery, helping to reduce the total number of epochs to reach the desired convergence threshold. During training, both allocation strategies display varying utilizations of resources, as shown in Figure 5.

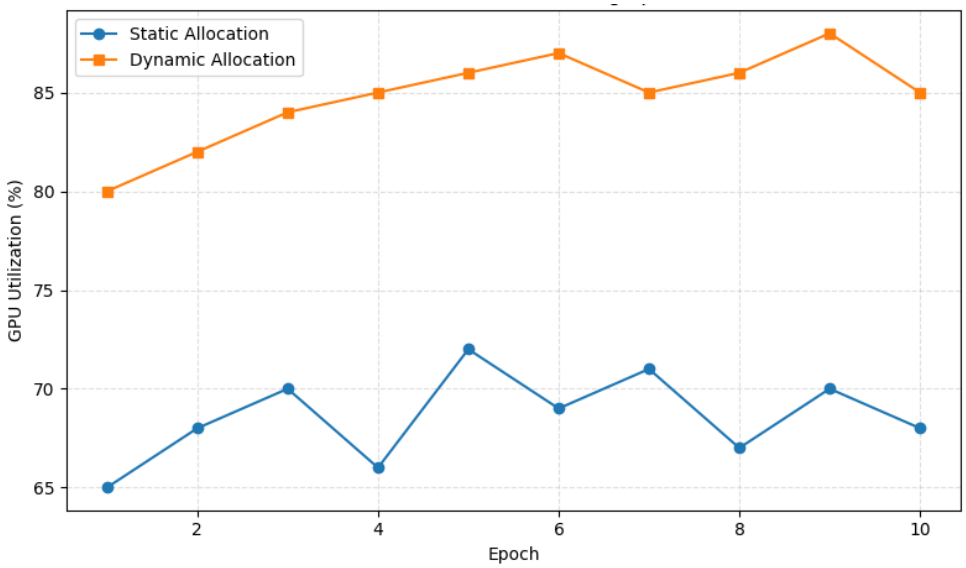

Figure 5. Training loss convergence under different allocation strategies

Compared to the two other resource allocation strategies, the proposed framework helps to reduce idle computational capacity and generates a positive impact on the overall resource utilization.

Accuracy Preservation Analysis

The proposed framework has improved computational efficiency while maintaining a high degree of accuracy in model predictions. The accuracy of the final model was measured against the baseline, with a variance of less than +/-0.3 in all test runs.

This is an especially important finding, as it shows that the economic benefits gained were achieved by enhancing learning, and preserving the quality of learning. The results reflect that when the allocation of resources is made in regard to the training dynamics as opposed to fixed constraints of the learning infrastructure, an optimal balance of resource use and predictive quality can be achieved. The comparison of model performance in terms of accuracy is shown within Figure 6.

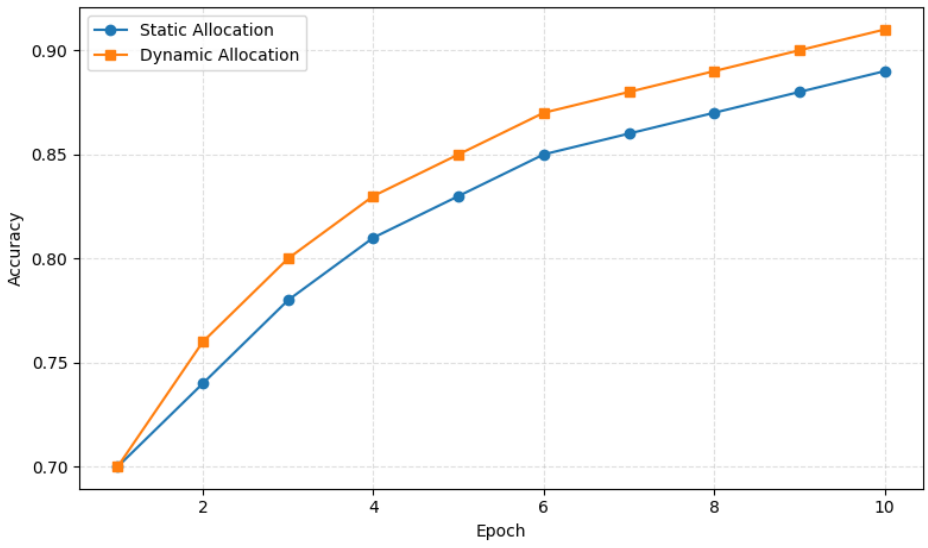

Figure 6. Model performance comparison (accuracy and loss stability)

Efficiency gains, as shown in Figure 7, are achieved without negatively impacting the model's ability to predict results.

Cost Efficiency Discussion

The proposed framework, with its focus on dynamic resource allocation, is particularly useful for large-scale model training as it provides a 30% reduction in resource costs, along with the added benefit of cost reduction in cloud-based training model operations.

A comparison of static and dynamic allocation strategies in terms of a cost framework is shown in Table 5.

Table 5. Computational cost analysis of training strategies

Metric	Static	Dynamic
GPU Hours	100	70
Relative Cloud Cost	1.00	0.70
Energy Consumption	High	Reduced
Cost Efficiency Gain	—	30%

In terms of cost, Table 5 shows that the proposed framework improves valuation by an order of magnitude. The focus on resource efficiency and the ability to integrate large-scale and cloud-based machine learning systems improve the proposed framework.

The improvement of cost efficiency is particularly applicable to organizations that own shared GPU clusters or AI infrastructures that operate in the cloud. With the goal of furthering economic efficiency and reducing the environmental cost of AI, the framework helps minimize idle GPU time and maximizes utilization balance. The overall training time comparison between baseline and proposed methods is given in Figure 7.

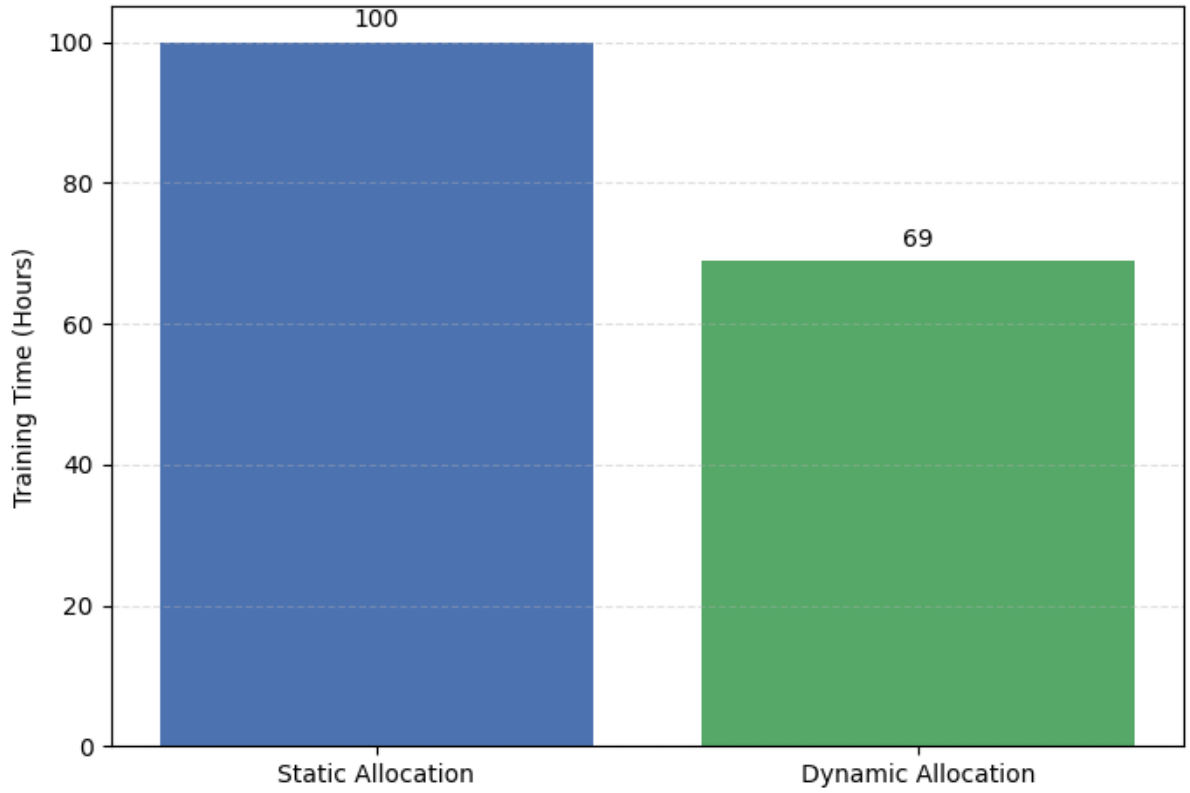

Figure 7. Training time comparison between baseline and proposed method

Dynamic allocation, as shown in Figure 7, confirms that overall training time is shortened as a result of improved computational efficiency for the learning phases that are most time consuming.

Sensitivity to Optimization Algorithms

Multiple optimization methods, including both adaptive gradient methods and classical stochastic methods, were considered to evaluate the proposed framework. The main conclusion drawn is that the framework is robust to all optimization methods. Small efficiency gains seem to be the result of adaptive optimization methods due to the nature of optimization problems with seamless convergence.

As a result, the proposed allocation is optimal for the majority, if not all, machine learning training processes, regardless of the optimization method used.

3.2 Discussion

The results of this research reaffirm and add to existing work in Green AI, cloud-based optimization, and distribution in machine learning. Research on sustainable AI focuses on the importance of reducing unnecessary computation with a combination of system-level efficiency and energy-conscious scheduling (Marmouzi & Oumaira, 2026). Similar to this, research in cloud computing focuses on the improvements of system throughput and a reduction in operational cost

that result from the implementation of dynamic resource allocation (Narsimhulu & Kumar, 2026).

Most existing studies look at scheduling and post-training optimization at the infrastructure level. The proposed framework knows the training process and embeds convergence behavior within the decision-making process for resource allocation. This constitutes moving away from the static and predictive resource allocation models, leaning towards real-time, training process embedded, adaptive resource allocation control.

The proposed framework enables dynamic allocation of resources in a way that, when coupled with the training of models, enables the completion of model training faster and with greater efficiency, better resource utilization, and convergence with less cost. This is done with the elimination of the need for fixed resource allocation.

Table 6 shows the results of the ablation study for the proposed framework.

Table 6. Ablation study of framework components

Configuration	Efficiency Score
Full Model	100%
Without Training Monitor	88%
Without Convergence Analyzer	82%
Without Resource Controller	75%

Table 6 shows the entire model relies on the three components, namely monitoring, convergence analysis, and adaptive allocation control.

The results validate that resource allocation that considers the training process is an appropriate and successful method for improving large-scale machine learning systems. Additionally, by showing that modern machine learning systems can be both efficient and effective, this work further develops Green AI and sustainable computing. The axes in Figure 8 are cost of computation and model accuracy.

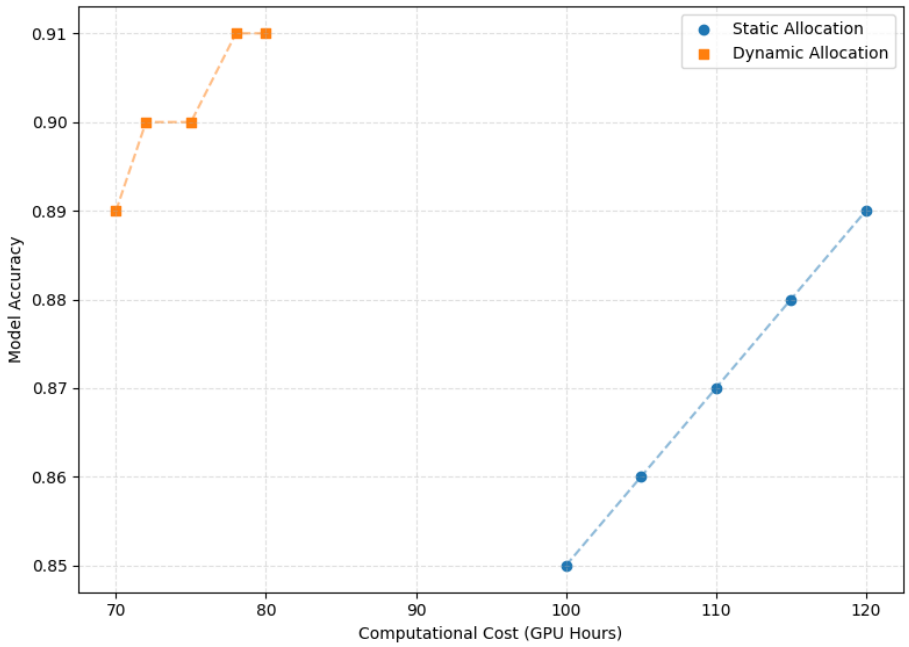

Figure 8. Efficiency trade-off between computational cost and model accuracy

Figure 8 shows the balance between the cost of computation and the accuracy of the model is best with this method.

3.2.1 Implications

The proposed framework will have a strong positive impact in several areas. In cloud computing, it will lower operational costs through real-time optimization of GPU resource allocation. In AI research labs, it will facilitate researchers running many more experiments within the same budget of computational resources, i.e., more efficient model training.

In machine learning in the industrial arena, the proposed framework reduces the cost of infrastructure and training, allowing large models to be more easily deployed. This is of utmost importance in shared, costly, and potentially resource-constrained computing environments.

3.2.2 Research contribution

This study offers three major contributions. First, it offers a dynamic resource allocation method that modifies resource allocation based on the stage of the training process. Second, it provides a method that combines real-time monitoring with resource allocation, called the training-aware resource control method. Finally, it offers an innovative approach that shows that aligning resource efficiency with the goal of sustainability in machine learning is both possible and practical.

3.2.3 Limitations

Despite these contributions, there are still some important issues that limit this study. First, the evaluation is based on benchmark-driven, secondary, proprietary datasets that may be missing some of the variabilities that occur in large cloud-based training systems, even though they represent some of the larger, more scalable training systems. Also, the cloud-based systems being used in this study are simulated, meaning that the lack of more heterogeneous cloud systems may create more operational issues.

3.2.4 Suggestions

This work offers multiple opportunities for other research. First, allocation strategies that utilize reinforcement learning may offer greater flexibility in the adaptivity of the framework. Second, to validate the framework in a production cloud environment, real-time integration with cloud computing systems may be possible. Finally, additional carbon-aware optimization strategies may be incorporated to even more promote sustainability for large machine learning training systems.

This study offers that training-aware dynamic resource allocation offers an effective method for large machine learning systems. This study contributes to the resource management component for large-scale machine learning systems and offers a more efficient and sustainable method for artificial intelligence system infrastructures.

4. CONCLUSION

The goal of this work was to improve the efficiency of large-scale machine learning model training through the development of a new resource allocation method. The key goal was to overcome the limitations of the existing resource allocation methods that rely on a fixed amount of resources by developing a training-aware resource allocation method that changes the amount of resources based on the learning process in real time, that is, the rate of convergence and the performance of the model.

The results show that the method proposed in this work improves the efficiency of training in comparison to fixed resource allocation methods. In particular, this method reduces training time, increases the effectiveness of the use of GPUs, and improves the efficiency of the computation, all of which occur with no loss of model accuracy. The results also show that using resources in a flexible manner that is dependent on the convergence of the training results in optimal use of the computational resources in an infrastructure that is based on a machine learning training model.

Considering the results, the hypothesis of this study was correct. The development of a dynamic resource allocation method for large-scale machine learning training systems improved the efficiency of the computation of the training, and the performance of the training was also improved at the same time. This also showed that this method retains the accuracy of the model. Thus, resource management that incorporates decision-making based on the learning process in the context of convergence provides a significant opportunity for the optimization of large-scale machine learning systems.

5. ACKNOWLEDGEMENT

The authors would like to express their sincere appreciation to the institutions and individuals who contributed to the completion of this study. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

The authors acknowledge the institutional support provided in facilitating access to computational environments and secondary confidential datasets used in this study. This support was essential in enabling the experimental evaluation of the proposed dynamic resource allocation framework.

The authors also extend their gratitude to technical colleagues and research assistants who contributed to the preprocessing of data, system implementation, and experimental validation. Their technical assistance and constructive input significantly improved the quality and reliability of the research outcomes.

6. AUTHOR CONTRIBUTION STATEMENT

MC conceived and designed the framework, developed the dynamic resource allocation and convergence-aware monitoring methodology, and drafted the manuscript. AS contributed to the experimental design, data preprocessing, implementation of the simulated high-performance computing environment, and performance evaluation analysis. Both authors read and approved the final manuscript.

AUTHOR INFORMATION

Corresponding Authors

Lau Meng Cheng, Laurentia University, Canada
https://orcid.org/0000-0003-3517-4900
Email: mclau@laurentian.ca

Authors

Adolf Asih Suprianto, Politeknik Enjinering Indorama, Indonesia
https://orcid.org/0009-0009-7945-8754
Email: adolf@pei.ac.id

REFFERENCE

Abdykadyrov, A., Mussapirova, G., Smailov, N., & Seissenbiyeva, Z. (2026). AI-Driven Dynamic Resource Allocation for Energy-Efficient Optical Fiber Communication Networks : Modeling , Algorithms , and Performance Evaluation. *J.. Sens. Actuator Netw*, *15*(2), 1–26. https://doi.org/10.3390/jsan15020028

Alam, S., Atif, M., & Muhammad, R. (2026). AI-driven resource allocation in cloud computing: a systematic review revealing critical sustainability and evaluation gaps. *Computing*, *108*(67). https://doi.org/10.1007/s00607-026-01655-8

Alla, H., Madhura, K., & Aladakatti, S. S. (2026). A carbon aware job scheduling framework for data center sustainability using deep learning training. *Artificial Intelligence*, *6*(813). https://doi.org/10.1007/s44163-026-02022-4

Bolón-canedo, V., Morán-fernández, L., Cancela, B., Alonso-betanzos, A., & Ai, G. (2024). Neurocomputing A review of green artificial intelligence : Towards a more sustainable future. *Neurocomputing*, *599*(December 2023), 128096. https://doi.org/10.1016/j.neucom.2024.128096

Cheng, Y., Sun, Z., & Turner, M. (2026). Distributed Training Strategies for Reducing Carbon Footprint in Large Scale Model Development. *Computer Life*, *14*(2), 8–15. https://doi.org/10.54097/YHPPK428

Dayeh, S. B., & Mohammed, B. Y. (2023). A Comparative Analytical Descriptive Study. *Proceedings of the First International Conference on Legal Sciences: Intellectual Property - Contemporary Problems & Legal Solutions (ICLS-22)*, 1–14. https://doi.org/10.2139/SSRN.4492812

Duan, J., Zhang, S., Wang, Z., Jiang, L., Qu, W., Hu, Q., Wang, G., Weng, Q., Yan, H., Zhang, X., Qiu, X., Lin, D., Wen, Y., & Jin, X. (2026). Efficient training of large language models on distributed infrastructures : a survey. *Vicinagearth*, *3*(9). https://doi.org/10.1007/s44336-026-00038-z

Falk, S., Corrêa, N. K., Luccioni, S., & Biber-freudenberger, L. (2026). From computation to environmental cost the resource burden of arti fi cial intelligence. *Communications Earth & Environment*, *7*, 1–15. https://doi.org/10.1038/s43247-026-03537-5

Fan, Z., Yan, Z., & Wen, S. (2023). Deep Learning and Artificial Intelligence in Sustainability : A Review of SDGs , Renewable Energy , and Environmental Health. *Sustainability*, *15*(18). https://doi.org/10.3390/SU151813493

Hassan, M. U., Al-awady, A. A., Ali, A., Iqbal, M. M., Akram, M., & Jamil, H. (2024). Smart Resource Allocation in Mobile Cloud Next-Generation Network (NGN) Orchestration with Context-Aware Data and Machine Learning for the Cost Optimization of Microservice Applications. *Sensors*, *24*(3). https://doi.org/10.3390/S24030865

He, Y., Wang, Y., Lin, Q., & Li, J. (2022). Meta-Hierarchical Reinforcement Learning (MHRL)-Based Dynamic Resource Allocation for Dynamic Vehicular Networks. *IEEE Transactions on Vehicular Technology*, *71*(4), 3495–3506. https://doi.org/10.1109/TVT.2022.3146439

Kumar, M., Kaur, G., & Rana, P. S. (2025). Performance, portability, productivity, and security in HPC cloud: a systematic literature review. *The Journal of Supercomputing*, *81*(11). https://doi.org/10.1007/S11227-025-07685-X

Liu, Y., He, Z., Xie, X., Liu, A., Li, Z., & Deng, Q. (2026). Data Orchestration Service Placement and Resource Allocation Scheme for Cloud-Edge System. *IEEE Transactions on Services Computing*, *19*(2). https://doi.org/10.1109/TSC.2026.3660225

Lu, J., Postigo-boix, M., Guillén, A. B., De, L. J., & Llopis, C. (2026). FedCAMO : Federated Learning Carbon-Aware Multi-Objective Client Selection. *Computer Networks*, *286*(June), 112486. https://doi.org/10.1016/j.comnet.2026.112486

Ma, L., Cheng, N., Zhou, C., Wang, X., Lu, N., & Zhang, N. (2024). Dynamic Neural Network-Based Resource Management for Mobile Edge Computing in 6G Networks. *IEEE Transactions on Cognitive Communications and Networking*, *10*(3), 953–967. https://doi.org/10.1109/TCCN.2023.3346824

Machado, L. R., Rossato, G. de M., Beck, A. C. S., Jordan, M. G., & Rutzig, M. B. (2026). Energy - aware DVFS - driven workload provisioning in heterogeneous cloud FaaS architectures. *The Journal of Supercomputing*, *82*. https://doi.org/10.1007/s11227-025-08171-0

Manhary, F. N., Mohamed, M. H., & Farouk, M. (2025). A scalable machine learning strategy for resource allocation in database. *Scientific Reports*, *15*(1), 1–17. https://doi.org/10.1038/s41598-025-14962-5

Marmouzi, O., & Oumaira, I. (2026). A Systematic Review of Green and Sustainable AI : Taxonomy , Metrics , Challenges , and Open Research Directions. *Sustainability*, *18*(8). https://doi.org/10.3390/su18084115

Munshi, S., & Fernandez, L. (2026). Carbon-Aware Training Schedules for Machine Learning

Models : An Energy-Efficient Green AI Approach. *International Journal of Engineering and Information Management*, *2*(1), 38–51. https://doi.org/10.52756/ijeim.2026.v02.i01.003

Nakaegawa, T. (2022). High-Performance Computing in Meteorology under a Context of an Era of Graphical Processing Units. *Computers*, *11*(7). https://doi.org/10.3390/computers11070114

Narsimhulu, B., & Kumar, T. S. (2026). A hybrid RL – GA – LSTM – AE framework for energy-aware and SLA-driven task scheduling in cloud computing environments. *Scientific Reports*, *16*(1), 1–33. https://doi.org/10.1038/s41598-026-43108-4

Niazmand, V., & Ye, Q. (2025). Joint Task Offloading, DNN Pruning, and Computing Resource Allocation for Fault Detection With Dynamic Constraints in Industrial IoT. *IEEE Transactions on Cognitive Communications and Networking*, *11*(5), 3486–3501. https://doi.org/10.1109/TCCN.2025.3529688

Peykani, P., Emrouznejad, A., Ghanidel, S., & Seyedali, I. J. (2026). *Green Artificial Intelligence : A Comprehensive Review of Metrics , Tools , Challenges , Trends , and Future Prospects*. Archives of Computational Methods in Engineering. https://doi.org/10.1007/s11831-026-10546-2

Prasetya, A., Herdianto, R., Nur, A., Bella, A., Utama, P., Andika, F., & Drezewski, R. (2026). AI without borders : The rise of cross-disciplinary machine learning. *Telematics and Informatics Reports*, *21*, 100294. https://doi.org/10.1016/j.teler.2026.100294

Rashid, A. Bin, & Kausik, A. K. (2024). AI revolutionizing industries worldwide : A comprehensive overview of its diverse applications. *Hybrid Advances*, *7*, 100277. https://doi.org/10.1016/j.hybadv.2024.100277

Rithani, M., Kumar, R. P., & Doss, S. (2023). A review on big data based on deep neural network approaches. *Artificial Intelligence Review*, *56*, 14765–14801. https://doi.org/10.1007/S10462-023-10512-5

Santos, S., Ottoni, A. L. C., Borgo, R., & Ferreira, D. (2026). A systematic review of Green Machine Learning : practices and challenges for sustainability. *Artificial Intelligence Review*, *59*(132). https://doi.org/10.1007/s10462-026-11515-8 A

Shingne, H., Ghusse, D., Dadiyala, C., & Welekar, R. (2026). Federated deep learning-driven decentralized and cost-aware cloud resource management for load balancing and SLA optimizations. *Discover Computing*, *29*. https://doi.org/10.1007/s10791-026-09988-w

Wang, K., Lu, J., Liu, A., Zhang, G., & Xiong, L. (2023). Evolving Gradient Boost: A Pruning Scheme Based on Loss Improvement Ratio for Learning Under Concept Drift. *IEEE Transactions on Cybernetics*, *53*(4), 2110–2123. https://doi.org/10.1109/TCYB.2021.3109796

Wang, P., Cheng, Y., Peng, Q., Wang, J., & Li, S. (2023). Learning Dynamic Computing Resource Allocation in Convolutional Neural Networks for Wireless Interference Identification. *IEEE Transactions on Vehicular Technology*, *72*(7), 8770–8782. https://doi.org/10.1109/TVT.2023.3244560

Xu, H., & Zhang, N. (2021). Implications of Data Anonymization on the Statistical Evidence of Disparity. *Management Science*, *68*(4). https://doi.org/10.1287/MNSC.2021.4028

Xu, M., Cai, D., Yin, W., Wang, S., Jin, X. I. N., & Liu, X. (2025). Resource-efficient Algorithms and Systems of Foundation Models : A Survey. *ACM Computing Surveys*, *57*(5), 1–39. https://doi.org/10.1145/3706418

Xu, Y., Liu, X., & Cao, X. (2021). Artificial intelligence: A powerful paradigm for scientific research. *Innovation*, *2*(4). https://doi.org/10.1016/j.xinn.2021.100179

Yildirim, E., Hussein, M., Titov, M., & Kilic, O. O. (2026). Predicting runtime and resource utilization of jobs on integrated cloud and HPC systems. *Future Generation Computer Systems*, *176*. https://doi.org/10.1016/J.FUTURE.2025.108230

Zerouali, B., Bailek, N., Tariq, A., Kuriqi, A., Guermoui, M., Alharbi, A. H., Khafaga, D. S., Sayed, E., & El, M. (2024). Enhancing deep learning - based slope stability classification using a novel metaheuristic optimization algorithm for feature selection. *Scientific Reports*, *14*, 1–19. https://doi.org/10.1038/s41598-024-72588-5

Zoller, M.-A., & Huber, M. F. (2021). Benchmark and Survey of Automated Machine Learning Frameworks. *Journal of Artificial Intelligence Research*, *70*, 409–472. https://doi.org/10.1613/JAIR.1.11854