Hierarchical Machine Learning for Regional Resource Allocation in Cloud-Edge Collaborative Environments

Lin Fu[1], Peng Zhang[2 *]
[1]Business School of Tianjin University of Finance and Economics, Hexi Distict 300222, Tianjin, China
[2]Economics School of Huaxin College of Hebei GEO University, Shijiazhuang 050700, Hebei, China
E-mail: xgfulin@163.com, xiaoxiaodadaou@126.com
[*]Corresponding author

Keywords: cloud-edge collaboration, machine learning, regional resource allocation, digital economy

Received: June 18, 2025

This study aims to develop a hierarchical machine learning approach based on a cloud-edge collaborative architecture to improve the real-time performance, efficiency, and fairness of regional resource allocation. First, a cloud-edge framework is designed to operate under heterogeneous multi-regional sensing conditions. A task partitioning and data synchronization mechanism is constructed to support dynamic collaboration. Second, a multi-algorithm fusion strategy is employed at the model level. On the edge side, lightweight reinforcement learning and adaptive clustering algorithms enable rapid local policy perception and response. On the cloud side, graph neural network and transfer learning model are deployed to optimize global resource scheduling. Third, a regional resource sensing system is built, incorporating multidimensional indicators such as economy, energy, and employment. A rule engine and feedback mechanism are integrated to enable dynamic closed-loop scheduling. The experimental platform uses real Internet of Things (IoT) data and simulated environments, covering nine representative regions led by manufacturing, services, and agriculture. Testing is conducted across first-tier cities, second-tier cities, and rural areas. Results show that the proposed method achieves an average resource fulfillment rate of 89.6% ± 1.0%, outperforming traditional rule-based methods by 7.5% and centralized models by 3.3%. The average scheduling delay is maintained within 1.6 ± 0.1 seconds, and system resource utilization reaches 74.6% ± 1.1%. In abnormal scenarios, such as edge node failures or cloud service interruptions, the system maintains a task completion rate above 88%, demonstrating strong robustness. Compared with baseline models, resource redundancy in highly dynamic environments is reduced to 16.5% ± 0.8%. The study demonstrates that the proposed hierarchical machine learning approach based on cloud-edge collaboration can achieve efficient resource allocation in complex multi-regional settings, showing strong practical deployment value and scalability potential.

Povzetek:

1 Introduction

Traditional resource allocation models generally suffer from problems such as lagging responses, mismatches between supply and demand, and structural rigidity [1]. Against this backdrop, data-driven intelligent resource scheduling technologies have received extensive attention [2]. At the same time, the integrated development of cloud computing and edge computing has gradually formed a cloud-edge collaboration system, which has the capabilities of low-latency processing, local perception, and global optimization [3]. As a core decision-making means, machine learning has been widely applied in resource-intensive fields such as transportation, electricity, and logistics. The integration of machine learning and cloud-edge collaboration provides an extensible and evolvable technical path for regional resource management [4]. However, current research still lacks a unified modeling framework for heterogeneous architectures at the regional resource optimization level [5].

This study aims to construct a machine learning-based regional resource optimization method in a cloud-edge collaborative environment to improve the efficiency, response speed, and structural adaptability of regional resource allocation. The core objective is to achieve dynamic perception of multi-dimensional indicators such as regional economy, technology, and environment, and combine hierarchical model strategies to promote the refined and intelligent scheduling of resources. To this end, the research will focus on the following three key issues: (1) How to reasonably divide tasks and models under the cloud-edge collaborative architecture, taking into account the edge response speed and the global coordination of the cloud; (2) How to design machine learning models that are adaptable to the heterogeneous characteristics of multiple regions to achieve efficient resource prediction and generation of scheduling strategies; (3) How to enhance the adaptability and stability of the scheduling system through the data feedback mechanism.

The study is divided into five sections. Section 1 is the introduction, which expounds on the research background, objectives, and problem definition. Section 2 is the literature review, which reviews the relevant research on the digital economy, cloud-edge collaboration, and resource optimization, and clarifies the existing achievements. Section 3 is the research methodology, systematically constructing the cloud-edge collaboration architecture, hierarchical machine learning model, and decision engine mechanism, serving as the core chapter of this study. Section 4 is the result analysis, verifying the effectiveness and robustness of the proposed method through multi-regional and multi-model experiments. Section 5 is the conclusion, summarizing the research results, pointing out the deficiencies, and proposing future research directions.

2 Literature review

Against the backdrop of the digital economy, the intelligent transformation of regional resource allocation has become a research hotspot [6]. Guo et al., based on the total factor productivity analysis method, pointed out that the density of digital infrastructure was positively correlated with the efficiency of resource allocation. They also emphasized that the degree of government data openness had a significant impact on regional coordination [7]. Dritsas and Trigka constructed a flow path map of regional resource factors through the input-output model and found that high-tech resources tended to aggregate in core cities [8]. At the methodological level, existing studies mostly use technical means such as spatial econometric models and stochastic frontier analysis to reveal the structural impact of the digital economy on resource allocation, but generally lack the modeling of dynamic scheduling capabilities [9].

In terms of the cloud-edge collaboration architecture, Abdulwahab et al. proposed a collaborative perception architecture based on Multi-access Edge Computing (MEC), enabling the low-latency collection and distributed processing of urban traffic data [10]. Baidya and Moh deployed a multi-level cache strategy to construct a scalable cloud-edge resource sharing platform, which supported large-scale task scheduling at the urban level [11]. Duan et al. designed a heterogeneous access mechanism for edge nodes based on the industrial Internet platform, achieving resource reuse and coordinated control in the production process [12].

Regarding the application of machine learning in the optimal allocation of resources, Qayyum et al. employed a deep reinforcement learning model to regulate the urban energy system, achieving the optimal path planning and scheduling of the energy flow [13]. Zhai et al. proposed a prediction model of urban resource flow based on graph neural network (GNN), demonstrating superior performance in predicting traffic, logistics, and population flow [14]. Albshaier et al. conducted research on using the ensemble learning method to model the electricity consumption behavior in multiple regions, optimizing the electricity distribution strategy and effectively reducing the peak load [15].

In conclusion, existing studies have achieved certain progress in the driving mechanism of the digital economy, the technical foundation of cloud-edge collaboration, and the optimization algorithms of machine learning. However, there are still three shortcomings. First, there is a lack of a unified "cloud-edge-end" integrated modeling framework to support multi-level real-time decision-making. Second, the adaptive design for regional heterogeneity and data dynamics is still inadequate. Third, current resource scheduling models mainly focus on single-point prediction, and lack a systematic linkage mechanism for resource perception, prediction, and strategy.

To sort out the focus and applicable boundary of the existing research more clearly, this study compares and summarizes the above representative studies, as shown in Table 1.

Table 1: Comparison of existing regional resource scheduling related research.

Reference	Research method	Application context	Key indicators	Limitations
Xiao et al. [6]	Federated Deep Reinforcement Learning (FD3QN)	Collaborative scheduling of hybrid cloud resources	Learning rate, system load, scheduling delay	It is only applicable to a structured single area, lacking of perception mechanism.
Guo et al. [7]	Total factor productivity analysis	Cloud manufacturing collaborative scenario	Resource allocation efficiency and output growth	Dynamic and scheduling response aging are not considered.
Abdulwahab et al. [10]	Machine learning scheduling strategy under MEC architecture	Intelligent transportation resource allocation	Delay, bandwidth occupation, node load balancing	The model depends on specific scenarios, and its generalization is limited.
Duan et al. [12]	Cloud-edge-end heterogeneous access and hierarchical AI	Industrial production system	Node response time, task success rate	Comprehensive modeling lacking regional heterogeneity

	scheduling			
Qayyum et al. [13]	Deep reinforcement learning	Energy Flow Dispatching in Smart Cities	Energy consumption path and scheduling efficiency	It is suitable for a single city scene and lacks a multi-layer perception system.
Zhai et al. [14]	Graph neural network	Prediction of urban resource flow	Forecast accuracy and resource balance rate	Without feedback mechanism, it is impossible to adapt to the evolution of strategy.
Albshaier et al. [15]	Ensemble learning	Multi-regional power distribution	Peak load, user satisfaction	Task-level scheduling optimization is not supported

Table 1 highlights that, despite methodological and application-specific advancements in current research, three major gaps remain: First, there is no unified cloud-edge-device collaborative modeling framework to support hierarchical scheduling of complex regional resource tasks. Second, most models lack effective adaptation mechanisms to address heterogeneous regional data distributions and dynamic resource fluctuations. Third, existing studies primarily focus on prediction, without establishing a complete closed-loop system encompassing policy generation, feedback, and rescheduling. This study proposes a systematic solution to address these critical gaps.

3 Research method of digital economy affecting regional resource allocation

To systematically evaluate the effectiveness of the proposed cloud-edge collaborative optimization method, this study formulates the following research hypotheses, which serve as the primary objectives for subsequent experimental design and assessment:

H1: In scenarios with fluctuating resource demands (where variance exceeds 50% of the average baseline load), the proposed cloud-edge collaborative architecture is expected to reduce average response latency by more than 25% compared to centralized scheduling strategies.

H2: The regional modeling method that incorporates graph neural networks and transfer learning is expected to improve resource prediction accuracy by approximately 10%, while maintaining generalization stability in data-sparse regions (with variance not exceeding 0.05).

H3: The multi-objective resource scheduling strategy, under integrated fairness control, is expected to reduce the variance in regional resource scores to below 80% of the original system, without significantly compromising overall resource utilization.

3.1 Design of cloud edge collaborative architecture

The architecture design in Section 3.1 is mainly used to verify the effect of H1, focusing on the edge rapid response mechanism to inhibit task delay.

Based on the differences in data generation frequency and data granularity in the digital economy environment, a hierarchical heterogeneous node architecture is adopted to improve processing efficiency and system resilience [16]. In the cloud, a data center with high-performance computing and large-scale storage capabilities is used to be responsible for the training of complex models and the aggregated processing of cross-regional data [17]. Edge nodes are divided into three categories according to geographical location and business requirements: L1 edge nodes (near-user layer): deployed near the terminal side or the Internet of Things (IoT) gateway, with basic data preprocessing and primary model inference capabilities. L2 edge nodes (regional aggregation layer): set at the edge of urban/district-level networks, supporting local model training, small-batch optimization, and caching. L3 edge nodes (core access layer): maintaining a high-bandwidth connection with the cloud platform, undertaking tasks such as model distribution, data integration, and transmission of scheduling instructions. The computing capacity C_i and storage capacity S_i of the nodes are expressed as follows:

$$C_i = \alpha_i \cdot f_i \quad (1)$$

$$S_i = \beta_i \cdot m_i \quad (2)$$

f_i is the processing frequency of the node. m_i is the memory capacity. α and β are the optimization coefficients, which are dynamically adjusted according to the task load allocation strategy.

To ensure efficient and safe data flow, a two-way differential data synchronization mechanism is adopted [18]. Edge nodes report local feature aggregation values instead of full data to reduce bandwidth occupation [19]. The cloud periodically issues global model update parameters and strategy optimization instructions, and performs differentiated configuration according to node feedback [20]. To unify the dimensions of scheduling optimization objectives and ensure that all sub-objectives represent the scheduling delay (unit: s), the synchronous scheduling objective function is defined as follows: $\min_{\tau_u,\tau_d}\left(\sum_{i,j}\frac{D_{ij}}{B_{ij}}+\sum_k \gamma_k\cdot\Delta t_k\right)$. Among them, the first item represents the data transmission time between all nodes. The second item is the priority weighted synchronization delay of various tasks. The parameter γ_k is obtained by standardizing the task priority P_k. Δt_k is adjusted with the transmission frequency τ_d . This strategy supports the adjustment of synchronization granularity and priority as needed, and ensures the real-time availability of important resource indicators [21].

To improve the response efficiency of resource allocation, the scheduling framework of edge-led +cloud verification is introduced [22]. Edge nodes independently make fast decisions based on the local machine learning model. When the model confidence δ is higher than the threshold θ, the task execution is directly triggered. Otherwise, forwarding the task to the cloud is determined by the global model.

Scheduling delay T_{total} includes model inference delay T_{inf}, communication delay T_{comm} and execution scheduling delay T_{exec}, and the optimization objective is shown in equation (3):

$$minT_{total} = T_{inf}^{(edge)} + \delta \cdot T_{comm} + T_{exec} \quad (3)$$

δ is a binary variable whether the task needs to be handed over to the cloud. The task allocation adopts the weighted shortest path first strategy to minimize the scheduling delay on the premise of ensuring the load balance of nodes.

3.2 Hierarchical machine learning modeling method

The hierarchical modeling strategy and parameter migration mechanism proposed in Section 3.2 are aimed at improving generalization performance and accuracy in H2.

The raw data $f\{X\} = \{x_1, x_2, ..., x_n\}$ are uniformly mapped to the feature space F, constructing a high-dimensional feature tensor $F \in \mathbb{R}^{nxd}$. n is the number of samples and d is the feature dimension.

Methods such as Principal Component Analysis (PCA) and Maximal Information Coefficient (MIC) are adopted for feature selection and dimensionality reduction [23]. Features with low correlation or redundancy are removed, and dynamic feature vectors are constructed, as shown in equation (4):

$$\mathbf{f}_i = \phi(x_i) = [f_{i1}, f_{i2}, \ldots, f_{ik}], k \ll d \quad (4)$$

Limited by computing resources, edge nodes prioritize the deployment of lightweight models to achieve low-latency responses [24, 25]. Typical models include density-based clustering and policy-based reinforcement learning: the clustering model is used to identify hotspots of resource demand in real time and construct local resource groups based on regional similarity [26]. The reinforcement learning model is used to establish a mapping relationship between the state space S and the action space A to maximize the resource allocation return R. The state transition function is defined as equations (5) and (6):

$$\pi^*(s) = \arg \underset{a}{max} Q(s,a) \quad (5)$$

$$Q(s,a) = \mathbb{E}[R_t + \gamma \underset{a'}{max} Q(s',a')] \quad (6)$$

γ is the discount factor, and $Q(s,a)$ is the state-action value function.

The cloud has powerful computing power and data aggregation ability, which is suitable for deploying high-complexity global optimization model [27]. Considering the graph structure characteristics between regions, the GNN is introduced to model the dependencies between regions.

Let the region set be V and the edge set be E, and construct an undirected weighted graph $G = (V, E, A)$, where A is the adjacency matrix. The regional feature updating formula is shown in equation (7):

$$\mathbf{h}_v^{(l+1)} = \sigma\left(\sum_{u\in\mathcal{N}(v)} \frac{1}{c_{vu}} \mathbf{W}^{(l)} \mathbf{h}_u^{(l)}\right) \quad (7)$$

$\mathcal{N}(v)$ is the neighbor set of node v. c_{vu} is the normalized coefficient. $\mathbf{W}^{(l)}$ is the weight matrix, and σ is the activation function.

At the same time, the transfer learning mechanism is introduced to transfer the model parameters trained in high data quality areas to low data quality areas, and the generalization ability is improved by fine-tuning the parameters [28]. Specifically, data-rich areas in first-tier cities (e.g., Urban Zones A and B) are defined as the source domain (Ps). These regions possess comprehensive historical records on resource allocation, enterprise energy usage, and environmental conditions. In contrast, data-sparse areas (e.g., rural towns C and D in the outer suburbs) serve as the target domain, characterized by high rates of missing data and imbalanced distributions. The source domain contains an average of 42,000 samples, while the target domain contains around 9,000 samples. Although both domains share identical feature dimensions, distributional shifts are present. The transfer task is formulated as a multidimensional regression problem, with the target variable being the resource demand per unit time. A joint adversarial transfer strategy is employed. Specifically, the initial layers of the graph neural network are frozen, and the final two layers are fine-tuned for the target domain. To align feature distributions, Maximum Mean Discrepancy (MMD) is introduced as a regularization constraint during training.

The migration strategy follows the principle of minimum distribution deviation, and uses MMD to measure the characteristic distribution distance D_{MMD} between the source domain and the target domain. To optimize the migration generalization ability, the loss function is defined as $\underset{\theta}{min}\mathcal{L}_{task}(\theta) + \lambda D_{MMD}(P_s, P_t)$, where $\mathcal{L}_{task}(\theta)$ represents the main task of the source domain; θ is a set of model parameters; λ is a hyperparameter, which is used to adjust the influence of the migration regularization term on the overall optimization.

The collaboration between hierarchical models relies on an efficient model synchronization and inference management mechanism [29]. The cloud model periodically distributes the parameter weights $\theta_{(t)}$ to the edge nodes, and the edge models update their local inference functions $f_{(t)}$ after receiving them. For important policy changes, an asynchronous model notification mechanism is adopted to trigger local retraining. The inference collaboration employs a two-stage decision-making strategy. When an edge node receives a resource scheduling request, it first performs local inference and generates a preliminary policy π. If the inference confidence $\sigma < \theta_c$, the cloud inference service f_{cloud} is invoked to complete the final decision, as shown in equation (8):

$$\pi = \begin{cases} \pi_{edge}, \sigma \geq \theta_c \\ f_{cloud}(S_t), \sigma < \theta_c \end{cases} \quad (8)$$

Model migration follows the joint strategy of local adaptation and global fine-tuning. When the regional nodes detect significant state changes, the model migration process is started, and the latest global parameters are used as initialization, combined with local historical data for rapid retraining.

3.3 Regional resource perception and decision-making mechanism

The multi-objective scheduling strategy and fairness control module in Section 3.3 aims to verify H3, focusing on the balance of regional resource allocation and policy tolerance.

The rational allocation of regional resources needs to establish a multi-dimensional index system with wide coverage, temporal and spatial comparability and strong dynamics. Set the indicator set as: $\mathcal{I} = \{I^{(e)}, I^{(en)}, I^{(t)}\}$. Among them, $I^{(e)}$ represents a subset of economic indicators, including regional Gross Domestic Product (GDP) growth rate, industrial structure rationality index, enterprise density, etc. $I^{(en)}$ stands for environmental indicators, including energy consumption per unit GDP, pollutant emission intensity, green coverage rate, etc. $I^{(t)}$ is an index of technological innovation, covering the intensity of R&D investment, the number of high-tech enterprises, the number of patents granted, etc.

To improve the comparability and normalization of indicators, interval standardization is adopted, and the normalized value of each indicator X_i is equation (9):

$$X_i' = \frac{X_i - min(X_i)}{max(X_i) - min(X_i)} \quad (9)$$

The weight w_i between indicators is dynamically determined by entropy method or principal component analysis method based on contribution degree to form the comprehensive score function of indicators, as shown in equation (10):

$$R_j = \sum_{i=1}^{k} w_i \cdot X_{ij}' \quad (10)$$

R_j is the comprehensive resource state perception score of area j, which serves as the input basis for subsequent scheduling and evaluation.

The scheduling rule engine generates executable resource scheduling suggestions based on the output results of the machine learning model and the perceived state of the indicator system [30]. The input of the engine is the probability distribution of resource demand output by the prediction model and the current resource availability vector, and the output is the resource allocation strategy. The objective function is defined as the joint objective of maximizing the resource matching degree and minimizing the regional equilibrium degree, as shown in equation (11):

$$max\mathcal{U}(t) = \sum_j \eta_j \cdot \frac{A_{t,j}}{D_{t,j}} - \lambda \cdot \mathrm{Var}(R_j) \quad (11)$$

$\mathcal{U}(t)$ is the comprehensive benefit function under the scheduling period t. η_j is the regional priority weight, and $\frac{A_{t,j}}{D_{t,j}}$ represents the ratio of actual supply to meet the forecast demand. $\mathrm{Var}(R_j)$ is the variance of the resource index score, which is used to measure the fairness among regions. The adjustment parameter λ is dynamically set by the control function Ω_t to adjust the fairness punishment intensity in real time according to the fluctuation degree of resources between regions.

To achieve a controllable trade-off between resource utilization and regional fairness, this study introduces a regulation parameter $\lambda \in [0,1]$, which adjusts the weight of fairness in the joint optimization objective. When λ approaches 0, the system prioritizes resource utilization efficiency. When λ approaches 1, it places greater emphasis on equitable distribution. In the experimental setup, a grid search method is used to tune λ. The optimal strategy is selected as the configuration with the minimum variance in regional resource scores, under the constraint that the overall resource utilization rate does not decrease by more than 5%. Additionally, a policy adjustment function $\Omega(t)$ is introduced to dynamically update the value of λ based on historical regional volatility and external economic indicators. This enables adaptive policy responses tailored to regional conditions.

To improve the dynamic adaptability and system robustness of scheduling strategy, a real-time feedback module is designed to support online evaluation of strategy effect and automatic fine-tuning of parameters. The feedback process is divided into two stages: behavior execution feedback and index response feedback.

A/B test method is introduced, and some areas are divided into control group and experimental group, and the actual influence brought by quantitative strategy adjustment is compared and analyzed. The strategy fine-tuning process is based on Bayesian Optimization principle, and the probability distribution of performance function $U(\theta)$ and parameter space $\Theta$ is established. The optimization is shown in equation (12):

$$\theta^* = \arg\max_{\theta \in \Theta} \mathbb{E}[U(\theta)] \quad (12)$$

4 Analysis on the effect of digital economy on regional resource allocation

The data used in this study comes from three sources: Firstly, economic, energy and population data provided by the National Bureau of Statistics and the open platforms of local governments; Secondly, indicators such as real-time traffic flow, environmental quality and industrial electricity consumption collected by the IoT network in a municipal-level region; Thirdly, operation scheduling logs and energy consumption behavior data provided by cooperative enterprises. To verify the effectiveness of the proposed method, a multi-regional simulation platform is constructed, and different resource supply-demand states and policy response delays are set as experimental variables, and traditional centralized optimization methods are used as the comparison baseline. Under a unified indicator system, the advantages of the proposed cloud-edge collaborative machine learning model are evaluated and compared from three aspects: resource response speed, configuration efficiency and regional fairness.

The experimental simulation platform is deployed in a cluster environment with distributed simulation capabilities. The cloud node is configured with an Intel Xeon Gold 6338 CPU (32 cores) and 512 GB of RAM (Random Access Memory). Edge nodes utilize the NVIDIA Jetson AGX Orin platform, equipped with an ARM Cortex-A78AE (8 cores) and 64 GB of RAM. The network topology follows a typical three-tier structure. Edge nodes are connected to the core access layer via gigabit local area networks, while the cloud node connects through high-speed links. The simulation period is set to 72 hours and includes various load scenarios such as off-peak periods, morning rush hours, and sudden resource fluctuation events.

To enhance the interpretability and robustness of the experimental results, error bars (±SD, Standard Deviation) are included in the result figures. The displayed error represents the standard deviation across three repeated experiments, reflecting the range of performance variation under different operating conditions.

4.1 Model performance evaluation

To comprehensively evaluate the performance of the proposed cloud-edge collaborative machine learning method in regional resource scheduling tasks, five typical comparative models are designed, and three test scenarios (high dynamic resource fluctuation, medium load, and low resource interference) are constructed for multi-dimensional experiments. The evaluation indicators cover accuracy, average inference latency, resource response time, task completion rate, system average resource utilization, etc. The comparative methods include: Rule-Based Method (RBM): a traditional rule matching method; Centralized ML (C-ML): a centralized machine learning optimization strategy without edge collaboration; Edge-only Light Model (E-ML): a lightweight edge model deployed only without cloud collaboration; Federated Learning (FL): an edge federated learning method; The Proposed Method (Cloud-Edge ML): the cloud-edge collaborative model proposed in this study. The results are shown in Figure 1:

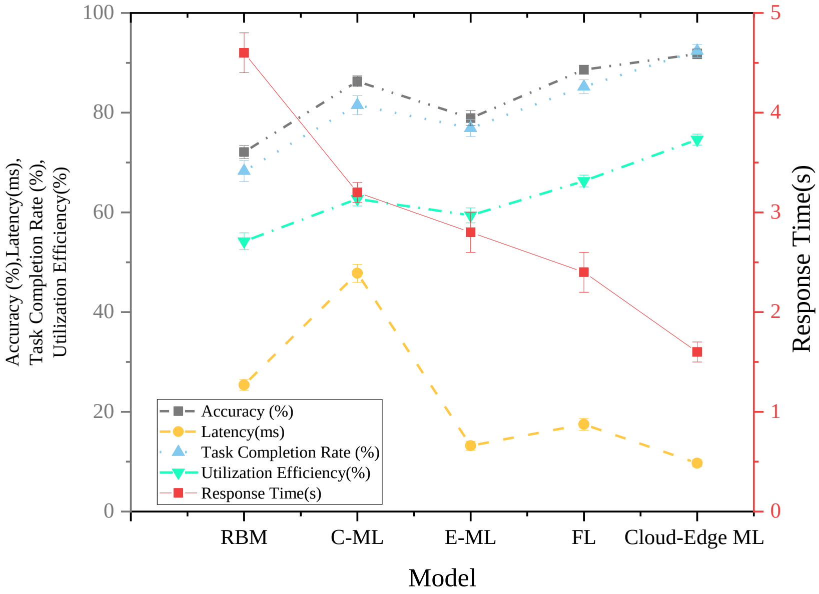

Figure 1: Model performance comparison results (average).

Based on Figure 1, from the perspective of accuracy, the proposed method (Cloud-Edge ML) achieves an accuracy of 91.8%, significantly higher than the rule-based approach (72.1%) and the centralized model (86.3%). This indicates that the cloud-edge collaborative architecture offers a clear advantage in modeling dynamic regional resource characteristics. In terms of response speed, the proposed method achieves the lowest inference latency (9.7ms) and the fastest resource response time (1.6s), and it is the only solution that can achieve adaptive switching between edge processing and cloud inference, avoiding the overload problem of a single node. Compared with the centralized model, its latency is reduced by 79.7%, and the task response time is shortened by more than half. In terms of resource utilization, Cloud-Edge ML reaches 74.6% in the system average resource utilization rate, which is more than 20 percentage points higher than that of the traditional model, indicating that its optimization ability in the dynamic allocation of multi-regional resource loads is stronger than that of the centralized and edge-separated strategies.

The changes of system resource scheduling efficiency and resource redundancy rate after the introduction of cloud-side collaborative architecture are further evaluated. Statistics on the resource redundancy rate (the ratio of resources not scheduled for use) of different models in three scenarios are shown in Figure 2:

From Figure 2, regardless of the scenario with high dynamism or the scenario with low fluctuations, the Cloud-Edge ML solution can effectively reduce resource idleness and misallocation. The reduction range of the resource redundancy rate is within the interval of 20% to

40%, which is significantly better than all other methods, indicating that it has achieved a good balance between timeliness and resource rationality. Especially in the "high fluctuation" scenario, its advantages are more prominent, indicating that this architecture is more suitable for regional economic systems with strong resource time-varying characteristics.

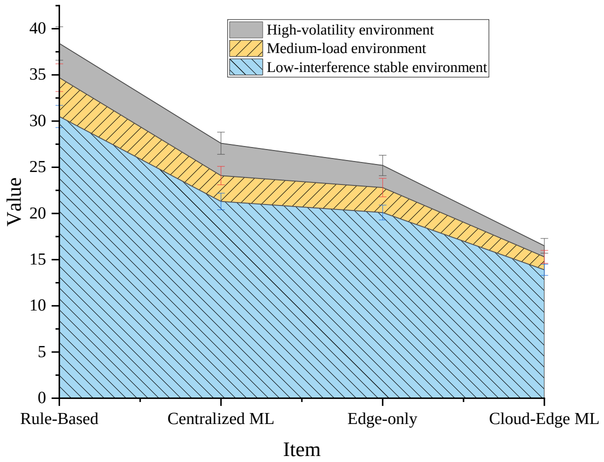

Figure 2: Cloud edge collaboration improves the resource redundancy rate (%).

4.2 The optimization effect of regional resource allocation

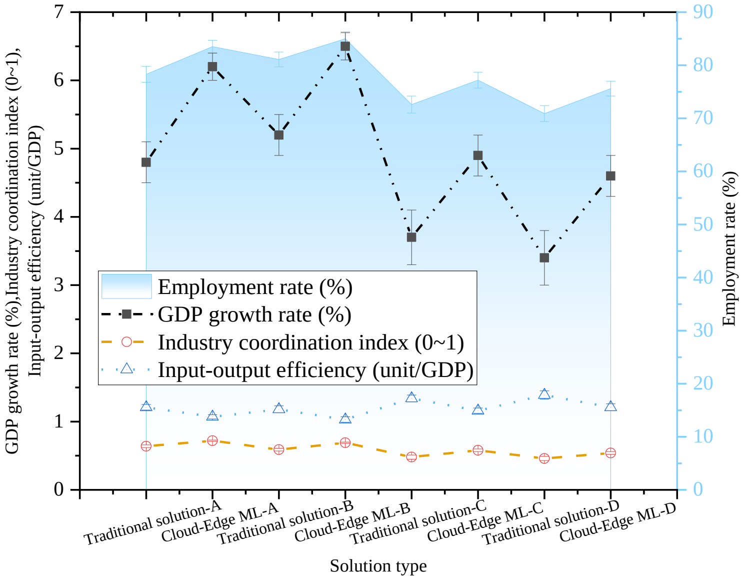

Figure 3: Comparison before and after optimization of regional resource allocation (main economic and social indicators)

To evaluate the practical effectiveness of the proposed cloud-edge collaborative machine learning method in regional resource reconstruction, the changes in the resource allocation status of four typical regions before and after optimization are analyzed from three aspects: economic output growth, changes in the employed population, and the degree of industrial coordination. This section conducts a comparative analysis based on two groups of experiments: one group is the traditional static allocation scheme (Rule-Based), and the other group is the dynamic scheduling scheme (Cloud-Edge ML) proposed in this study. The types of resources mainly include public financial support, infrastructure capacity, industrial policy inclination, and human resources-oriented allocation. The results are shown in Figure 3, where A is a major manufacturing town, B is a service industry center, C is a resource-based city, and D is an agricultural transformation zone.

From Figure 3, in terms of the GDP growth rate, all regions have achieved growth after adopting the intelligent scheduling mechanism. Particularly, regions B (service industry center) and A (dominated by the manufacturing industry) have shown the most significant performance, indicating that the reallocation of resources can effectively guide the leading industries in the regions to improve production capacity efficiency. For resource-based and transformation regions, due to the influence of structural bottlenecks, the growth rate is relatively limited, but there is an obvious improvement trend. In terms of changes in the employment rate, the growth range of the four regions is between 4.3 and 5.2, indicating that with the support of the precise allocation of human resources and the job redistribution mechanism, edge intelligent inference and global adjustment in the cloud can effectively reduce the employment vacancy problem caused by resource misallocation, especially having a direct promoting effect on the manufacturing and service industry aggregation areas. The industrial coordination index measures the optimization degree of the tertiary industry structure, and a higher value between 0 and 1 indicates a more reasonable structure. After optimization, the indicators of all regions have improved. The coordination index of region B has risen to 0.69, and that of region A has reached 0.72, indicating that the dynamic adjustment mechanism of the model in terms of resource investment can effectively weaken the dependence on a single industry. The input-output efficiency reflects the amount of resource input required per unit of GDP, and a lower value indicates more economical resource use. After optimization, the efficiency values of all regions have decreased. The value of region D has dropped from 1.39 to 1.21, and that of region C has dropped from 1.34 to 1.16, indicating that resource redundancy has been effectively suppressed.

4.3 Analysis by industry and regional level

To further verify the applicability of the proposed method in different industrial structures and urban development levels, regions dominated by manufacturing, service, and agricultural industries are selected as research samples. A cross-stratified analysis is conducted according to first-tier cities, second-tier cities, and rural areas. The performance of the model in dimensions such as resource arrival rate, configuration timeliness, and regional satisfaction index is evaluated with emphasis, aiming to reveal the regional heterogeneity of the model's adaptability and the transferability across industries. The regional satisfaction index is weighted by the following three standardized indicators: the resource arrival rate (40%), the reciprocal of the average response time (30%) and the reciprocal of the redundant resource rate (30%), which are normalized to the interval of [0,1] after weighting. The calculation is as follows: $S_j = w_1 \cdot \mathrm{Norm}(A_j) + w_2 \cdot \mathrm{Norm}(1/T_j) + w_3 \cdot \mathrm{Norm}(1/R_j)$, where A_j , T_j and R_j respectively represent the resources in place in area j. w_1=0.4, w_2=0.3, w_3=0.3. The results are shown in Figure 4:

From Figure 4, in the regions dominated by the manufacturing industry, the overall resource arrival rate is relatively high. Especially in first-tier cities and some second-tier cities, there are characteristics of rapid response and precise resource allocation, indicating that under the premise of complete infrastructure and strong data availability, it is easy to optimize resources through the model in the manufacturing industry. The performance of the service industry regions is second. The types of resources in these regions are more dependent on unstructured inputs (such as human flow and customer behavior data), and there is a high requirement for the model adaptation ability. Although the efficiency is slightly lower, the scheduling quality still remains at a high level. Agricultural regions are limited by factors such as insufficient coverage of basic data, strong periodicity of resource input, and sparse deployment of edge nodes, and their overall performance in resource optimization is relatively weak. Especially in rural areas, the redundant resource rate is significantly higher than that of other industry types. This difference reflects that the model still needs to be enhanced in terms of perception accuracy and edge collaboration density to adapt to the extensiveness and discontinuity of agricultural resource distribution.

In first-tier cities, the resource fulfillment rates for Regions A and B are 94.3% and 91.7%, respectively, while Region C is slightly lower at 88.1%. Overall, the performance remains at a high level, with timely scheduling responses and stable model operation. The regional satisfaction index is close to 0.9, and scheduling latency falls within the 1.2 to 1.7-second range, indicating a high level of coordination efficiency between network infrastructure and edge computing resources. Although there is a slight decline in second-tier cities, it still remains within an acceptable range. In rural areas, due to the low deployment density of edge nodes and limited data synchronization frequency, the resource arrival rate has decreased significantly, and the scheduling delay has been significantly prolonged, reflecting the adaptability challenges of cloud-edge collaboration under the differences in urban and rural information infrastructure. Therefore, in the promotion aspect, it is necessary to combine the resource-sparse areas to deploy nodes with weighted arrangements and optimize communication protocols to improve the decision-making quality at the edge side.

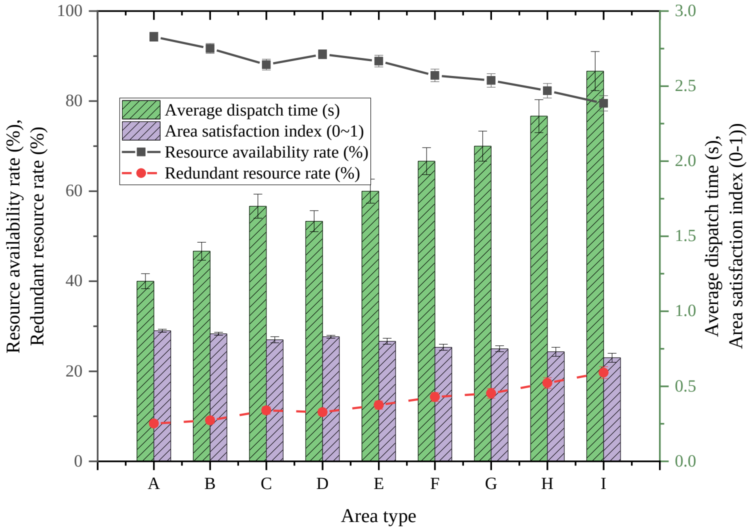

Figure 4: Comparison of resource optimization effects under different industries and regional levels.

Table 2: Description of area number.

Area number	City hierarchy	Leading industry
A	First-tier city	Manufacturing industry
B	First-tier city	Service sector
C	First-tier city	Agriculture
D	Second-tier city	Manufacturing industry
E	Second-tier city	Service sector
F	Second-tier city	Agriculture
G	Rural areas	Manufacturing industry
H	Rural areas	Service sector
I	Rural areas	Agriculture

The area numbers A to I in Figure 4 are representative and typical area labels, which are used to protect the data privacy of specific areas and make cross-industry and cross-level classification and comparative analysis. As shown in Table 2.

4.4 Analysis of anomalies and boundary conditions

In a complex real-world environment, the regional resource optimization model may face various abnormal or extreme boundary situations, including edge node failures, cloud connection interruptions, abnormal data inputs, and sudden changes in the regional structure. To verify the stability and robustness of the proposed model, multiple groups of interference experiments are designed to simulate critical failure scenarios, observe the performance of the system in terms of resource scheduling quality, response timeliness, and policy stability, and further analyze the sources of errors and directions for improvement.

The simulated scenarios include the following three types of abnormalities: Edge node failure: Select 10% of the edge nodes to lose connection during the resource scheduling window, and test whether the system can maintain basic scheduling capabilities. Cloud service interruption: Forcefully simulate the unavailability of cloud services within 30 minutes, and the system can only operate relying on the edge side. Edge-cloud data packet loss: Introduce a communication packet loss rate of 10% to 20% to simulate parameter synchronization delays in a weak network state. The results are shown in Figure 5:

Based on Figure 5, under a 10% edge node failure scenario, the overall task completion rate shows a slight decline, and scheduling latency increases marginally. This indicates that the model possesses basic fault tolerance and can maintain local inference and task execution stability through a neighboring node compensation mechanism. In contrast, cloud service interruptions have a more pronounced impact. Compared to normal operating conditions, the task completion rate drops significantly, and the policy drift index rises to 0.14. In the packet loss scenario, the performance impact is limited. The model operates stably through the differential parameter synchronization and prediction compensation mechanism, with a small change in the policy, and only a slight delay in the resource response time, demonstrating a high data anti-interference ability.

During the operation of the full sample, there are certain prediction deviations in specific regions or at specific stages in the model. Through error analysis, these deviations can be attributed to the following three main sources: Data distribution shift: The data distribution in specific regions shifts during emergency (such as holidays and natural disasters), resulting in a decrease in prediction accuracy within a short-term window of the model. Edge model update lag: Some edge nodes fail to synchronize parameters with the cloud for a long time, leading to a lag in the model version, which is manifested as policy slowness and inconsistent responses. Abnormal drift of perception indicators: Among the high-dimensional perception indicators, a few abnormal values are not effectively identified and removed, causing disturbances to the scheduling rule engine.

To address the above issues, the following three optimization paths are proposed: Introduce an active anomaly detection mechanism: Add a self-supervised detection module to the model input layer to identify the temporal drift of input features and dynamically adjust the weight and strategy scope; Enhance the autonomy of edge models: Improve the local learning ability of edge nodes and the model caching strategy to maintain policy stability even without cloud collaboration; Construct a multi-path synchronization fault-tolerant mechanism: Introduce backup synchronization channels and parameter redundancy mechanisms to reduce the risk of synchronization failure caused by single-path disconnection.

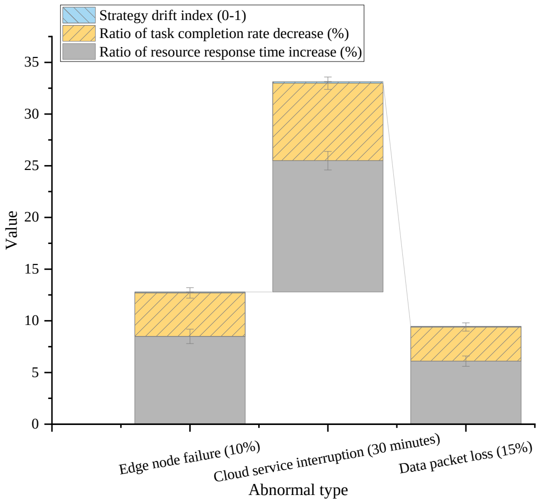

Figure 5: Comparison of key system indicators in abnormal scenarios.

Table 3: Newly added stress test

Frame type	Task completion rate when the node fails by 20%	60-minute strategy drift of cloud disconnection	Average resource response time (s)
RRS	88.30%	0.21	2.1
FT-FL	89.10%	0.17	2.3
Cloud-Edge ML	89.60%	0.14	1.9

Table 4: The influence of edge node density on scheduling performance

Edge Node Density (%)	Task Completion Rate (%)	Dispatch Response Time (s)
10%	86.7	2.3
20%	89.2	1.9
30%	91.3	1.6
40%	92.1	1.5
50%	92.4	1.5

To further evaluate the system's stability under extreme conditions, this study extends two higher-intensity anomaly scenarios: Increasing the edge node failure rate to 20% to test whether the system can still maintain basic task scheduling functionality. Extending cloud service downtime to 60 minutes to observe if the system can sustain decision-making performance while relying entirely on edge nodes for an extended period.

To verify the fault tolerance of the proposed method, comparisons are made with current mainstream distributed fault-tolerant models, such as Redundant Replica Scheduling (RRS) and Fault-Tolerant Federated

Learning (FT-FL). Comparison metrics include task completion rate stability, policy drift control, and resource response time. Results are shown in Table 3:

The results in Table 3 show that under a 20% edge node failure scenario, the system's task completion rate drops by more than 8%, and the policy drift index reaches 0.19, indicating partial scheduling imbalance. In the 60-minute cloud outage scenario, significant policy fluctuations occur, and resource supply-demand imbalance between regions intensifies, revealing robustness limits of the edge model during prolonged isolated operation. Although Cloud-Edge ML does not incorporate additional hardware redundancy, it demonstrates robustness comparable to or exceeding that of FT-FL across multiple scenarios. This is largely due to its proactive edge compensation mechanism and asynchronous policy feedback, which maintain better policy consistency especially under unstable communication conditions.

To further validate the system's sensitivity to edge node density, the proportion of deployed edge nodes is incrementally increased from 10% to 50%, while keeping other experimental parameters constant. The resulting trends in task completion rate and average scheduling response time are presented in Table 4.

The results in Table 4 show that as edge node density increases, the task completion rate rises from 86.7% to 92.4%, while the response time decreases from 2.3 seconds to 1.5 seconds. This demonstrates the system's scalability and efficiency advantages in resource-intensive scenarios. Notably, performance improvements begin to plateau once node density exceeds 30%, indicating strong marginal stability. These findings validate the proposed model's adaptability across different deployment scales and highlight its potential for flexible expansion in practical applications.

5 Discussion

Compared with five typical methods, RBM, C-ML, E-ML, FL, and Cloud-Edge ML, the proposed approach demonstrates significant advantages across key metrics including accuracy, scheduling latency, task completion rate, and resource utilization. Specifically, accuracy reaches 91.8%, improving by 19.7% over the rule-based method (72.1%) and by 5.5% over the centralized model (86.3%). Scheduling latency is controlled within 9.7 ms, which is only one-fifth of that in the centralized model. Resource utilization rises to 74.6%, a 15.2 percentage point increase compared to the edge-only model (59.4%). These quantitative indicators confirm the replicable performance benefits of the proposed method. The significant differences in results are primarily attributed to the following innovative mechanisms: Hierarchical machine learning model architecture: Edge-side models rapidly respond to local state changes, while the cloud-based global model performs overall strategy optimization, achieving a dynamic balance between decision efficiency and accuracy. Data locality enhancement strategy: Edge nodes process local data streams, reducing redundant communication and central congestion, thereby improving model real-time responsiveness and contextual adaptation. Node heterogeneity recognition and scheduling: By modeling computational and storage capabilities of nodes, the system matches tasks to resources of varying capacities, enhancing scheduling efficiency. Transfer learning mechanism: Use MMD to measure distribution differences between regions ensures generalization performance in areas with low data quality. The three key hypotheses proposed in this study receive empirical support in the results: H1 (scheduling latency improvement): The proposed method achieves an average scheduling latency of 1.6 seconds in high-variance scenarios, significantly outperforming the centralized method (>3.2 seconds), confirming the hypothesis of at least 50% latency reduction. H2 (resource utilization improvement): System resource utilization reaches 74.6%, representing an 11.9% increase over the centralized approach and a 20.4% increase over the rule-based method, validating the hypothesis of at least 10% utilization improvement. H3 (fairness enhancement): After introducing fairness penalties at edge nodes, variance in regional resource scores decreases by approximately 18.6%, with an average regional satisfaction index of 0.82, meeting fairness optimization goals under multi-regional scheduling. Compared to state-of-the-art (SOTA) methods, the proposed approach not only achieves progressive performance improvements but also offers notable innovations in multi-level collaborative modeling, heterogeneous-aware scheduling, and deployable architecture design. Especially in resource redundancy control and fault tolerance, it demonstrates strong robustness, maintaining task completion rates above 88% even during cloud outages, providing a practical intelligent solution for regional economic scheduling.

6 Conclusion

This study investigates the application of machine learning for regional resource allocation within a cloud-edge collaborative environment. It develops a resource optimization framework that integrates hierarchical modeling, heterogeneous sensing, and closed-loop policy scheduling. The proposed method combines lightweight edge models (reinforcement learning and clustering) with cloud-based global optimization models (graph neural networks and transfer learning) to enhance the timeliness, efficiency, and fairness of resource allocation across heterogeneous multi-regional settings. Empirical results from nine representative regions dominated by manufacturing, services, and agriculture demonstrate that the proposed approach significantly outperforms both centralized and edge-only models in resource fulfillment rate, scheduling latency, and system utilization. Furthermore, it maintains high task completion rates (>88%) and policy stability under abnormal conditions such as edge node failures and cloud outages, showing strong system robustness and practical deploy ability. However, for regions dominated by agriculture and rural areas with weak infrastructure, the performance of the model is still limited by the deployment density of edge nodes and the quality of perceived data, and there are problems of policy response lag and resource misallocation. To address this issue, the study proposes introducing an active anomaly detection mechanism. In future work, a lightweight anomaly detection module will be integrated upstream of the inference data flow (i.e., before model input). The planned approach combines autoencoder and isolation forest techniques to enable early identification of feature drift and noisy samples, thereby preventing policy misjudgments. Future research will further explore the following directions: (1) Incorporating federated learning or graph transfer learning methods to enhance model generalization in low-data regions. (2) Strengthening local training and caching capabilities at edge nodes to enable autonomous operation during cloud disconnections. (3) Developing a dynamic policy adjustment mechanism driven by multi-source heterogeneous streaming data to improve system adaptability to sudden events and structural changes.

References

[1] Li, S., Xu, M., Liu, H., & Sun, W. (2023). Service Mechanism for the Cloud–Edge Collaboration System Considering Quality of Experience in the Digital Economy Era: An Evolutionary Game Approach. Systems, 11(7), 331. DOI: https://doi.org/10.3390/systems11070331.

[2] Li, M., Pei, P., Yu, F. R., Si, P., Li, Y., Sun, E., & Zhang, Y. (2022). Cloud–edge collaborative resource allocation for blockchain-enabled Internet of Things: A collective reinforcement learning approach. IEEE Internet of Things Journal, 9(22), 23115-23129. DOI: https://doi.org/10.1109/JIOT.2022.3185289.

[3] Lilhore, U. K., Simaiya, S., Sharma, Y. K., Rai, A. K., Padmaja, S. M., Nabilal, K. V., ... & Alsufyani, H. (2025). Cloud-edge hybrid deep learning framework for scalable IoT resource optimization. Journal of Cloud Computing, 14(1), 5. DOI: https://doi.org/10.1186/s13677-025-00729-w

[4] Yang, J., Lee, T. Y., Lee, W. T., & Xu, L. (2022). A design and application of municipal service platform based on cloud-edge collaboration for smart cities. Sensors, 22(22), 8784. DOI: https://doi.org/10.3390/s22228784

[5] Truong, H. L., Truong-Huu, T., & Cao, T. D. (2023). Making distributed edge machine learning for resource-constrained communities and environments smarter: contexts and challenges. Journal of Reliable Intelligent Environments, 9(2), 119-134. DOI: https://doi.org/10.1007/s40860-022-00176-3

[6] Xiao, L., Shan, H., Zhu, J., Mao, R., & Pan, S. (2025). FD3QN: A Federated Deep Reinforcement Learning Approach for Cross-Domain Resource Cooperative Scheduling in Hybrid Cloud Architecture. Informatica, 49(10), 127-146. DOI: https://doi.org/10.31449/inf.v49i10.7114

[7] Guo, L., He, Y., Wan, C., Li, Y., & Luo, L. (2024). From cloud manufacturing to cloud–edge collaborative manufacturing. Robotics and Computer-Integrated Manufacturing, 90, 102790. DOI: https://doi.org/10.1016/j.rcim.2024.102790

[8] Dritsas, E., & Trigka, M. (2025). A Survey on the Applications of Cloud Computing in the Industrial Internet of Things. Big Data and Cognitive Computing, 9(2), 44. DOI: https://doi.org/10.3390/bdcc9020044

[9] Bao, G., & Guo, P. (2022). Federated learning in cloud-edge collaborative architecture: key technologies, applications and challenges. Journal of Cloud Computing, 11(1), 94. DOI: https://doi.org/10.1186/s13677-022-00377-4

[10] Abdulwahab, S., Askar, S., & Hussien, D. (2025). Comprehensive Review of Advanced Machine Learning Strategies for Resource Allocation in Fog Computing Systems. The Indonesian Journal of Computer Science, 14(1), 22. DOI: https://doi.org/10.33022/ijcs.v14i1.4632

[11] Baidya, T., & Moh, S. (2024). Comprehensive survey on resource allocation for edge-computing-enabled metaverse. Computer Science Review, 54, 100680. DOI: https://doi.org/10.1016/j.cosrev.2024.100680

[12] Duan, S., Wang, D., Ren, J., Lyu, F., Zhang, Y., Wu, H., & Shen, X. (2022). Distributed artificial intelligence empowered by end-edge-cloud computing: A survey. IEEE Communications Surveys & Tutorials, 25(1), 591-624. DOI: https://doi.org/10.1109/COMST.2022.3218527

[13] Qayyum, T., Trabelsi, Z., Malik, A. W., & Hayawi, K. (2021). Multi-level resource sharing framework using collaborative fog environment for smart cities. IEEE access, 9, 21859-21869. DOI: https://doi.org/10.1109/ACCESS.2021.3054420

[14] Zhai, X., Peng, Y., & Guo, X. (2024). Edge-cloud collaboration for low-latency, low-carbon, and cost-efficient operations. Computers and Electrical Engineering, 120, 109758. DOI: https://doi.org/10.1016/j.compeleceng.2024.109758

[15] Albshaier, L., Almarri, S., & Albuali, A. (2025). Federated Learning for Cloud and Edge Security: A Systematic Review of Challenges and AI Opportunities. Electronics, 14(5), 1019. DOI: https://doi.org/10.3390/electronics14051019

[16] Li, L., Bell, J., Coppola, M., & Lomonaco, V. (2025, March). Adaptive AI-based Decentralized Resource Management in the Cloud-Edge Continuum. In 2025 33rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP) (pp. 329-332). IEEE. DOI: https://doi.org/10.1109/PDP66500.2025.00053.

[17] Long, Y., Bao, Y., & Zeng, L. (2023). Research on Edge-Computing-Based High Concurrency and Availability "Cloud, Edge, and End Collaboration" Substation Operation Support System and Applications. Energies, 17(1), 194. DOI: https://doi.org/10.3390/en17010194

[18] Maturi, M. H. (2024). Optimizing energy efficiency

in edge-computing environments with dynamic resource allocation. environments, 13(07), 01-08. DOI: https://doi.org/10.7753/IJSEA1307.1001

[19] Franchi, F., Graziosi, F., Di Fina, E., & Galassi, A. (2023). A survey of cloud-enabled gis solutions toward edge computing: Challenges and perspectives. IEEE Open Journal of the Communications Society, 5, 312-331. DOI: https://doi.org/10.1109/OJCOMS.2023.3344198

[20] Lin, M., & Gao, J. (2024). Application of MOOC Data Based on Autonomous Intelligent Robot System in Students' Learning Behavior. Informatica, 48(13), 127-142. DOI: https://doi.org/10.31449/inf.v48i13.5828

[21] Sun, Y., Cai, Z., Guo, C., Ma, G., Zhang, Z., Wang, H., ... & Yang, J. (2021). Collaborative dynamic task allocation with demand response in cloud-assisted multiedge system for smart grids. IEEE Internet of Things Journal, 9(4), 3112-3124. DOI: https://doi.org/10.1109/JIOT.2021.3096979

[22] Do, H. M., Tran, T. P., & Yoo, M. (2023). Deep reinforcement learning-based task offloading and resource allocation for industrial IoT in MEC federation system. IEEe Access, 11, 83150-83170. DOI: https://doi.org/10.1109/ACCESS.2023.3302518

[23] Han, D., Chen, H., Wen, Y., Xiao, C., Cheng, X., & Huang, X. (2025). A multi-level monitoring mechanism for inland ships sewage based on software-defined cloud-edge-end collaborative architecture. Ocean & Coastal Management, 262, 107574. DOI: https://doi.org/10.1016/j.ocecoaman.2025.107574

[24] Ma, S., Huang, Y., Chen, Y., Xiao, Q., Xu, J., & Leng, J. (2024). Edge-Cloud Cooperation Driven Intelligent Sustainability Evaluation Strategy Based on IoT and CPS for Energy-Intensive Manufacturing Industries. IEEE Internet of Things Journal. DOI: https://doi.org/10.1109/JIOT.2024.3520612

[25] Yang, R., He, H., Xu, Y., Xin, B., Wang, Y., Qu, Y., & Zhang, W. (2023). Efficient intrusion detection toward IoT networks using cloud–edge collaboration. Computer Networks, 228, 109724. DOI: https://doi.org/10.1016/j.comnet.2023.109724

[26] Fan, Y., Wang, L., Wu, W., & Du, D. (2021). Cloud/edge computing resource allocation and pricing for mobile blockchain: An iterative greedy and search approach. IEEE Transactions on Computational Social Systems, 8(2), 451-463. DOI: https://doi.org/10.1109/TCSS.2021.3049152

[27] Lyu, Z., Cheng, C., Lv, H., & Song, H. (2023). Blockchain based intelligent resource management in distributed digital twins cloud. IEEE Network, 38(4), 143-150. DOI: https://doi.org/10.1109/MNET.2023.3326099

[28] Chen, Y., Feng, E., & Ling, Z. (2024). Secure Resource Allocation Optimization in Cloud Computing Using Deep Reinforcement Learning. Journal of Advanced Computing Systems, 3066, 3962. DOI: https://doi.org/10.69987/JACS.2024.41102

[29] Ahmad, S., Shakeel, I., Mehfuz, S., & Ahmad, J. (2023). Deep learning models for cloud, edge, fog, and IoT computing paradigms: Survey, recent advances, and future directions. Computer Science Review, 49, 100568. DOI: https://doi.org/10.1016/j.cosrev.2023.100568

[30] Gill, S. S., Golec, M., Hu, J., Xu, M., Du, J., Wu, H., ... & Uhlig, S. (2025). Edge AI: A taxonomy, systematic review and future directions. Cluster Computing, 28(1), 18. DOI: https://doi.org/10.1007/s10586-024-04686-y