



# Exploration of Data Center Cost Optimization Pathways Under Multi-generation CPU and GPU Collaborative Architectures

Yaqi Hou

School of Information Science and Engineering, Central South University, Changsha 410083, Hunan, China.

**How to cite this paper:** Yaqi Hou. (2026) Exploration of Data Center Cost Optimization Pathways Under Multi-generation CPU and GPU Collaborative Architectures. *Engineering Advances*, 6(1), 41-44.  
DOI: 10.26855/ea.2026.03.009

**Received:** January 10, 2026

**Accepted:** February 6, 2026

**Published:** March 3, 2026

\***Corresponding author:** Yaqi Hou, School of Information Science and Engineering, Central South University, Changsha 410083, Hunan, China.

## Abstract

In the operating context where multi-generation CPUs and GPUs are deployed in parallel for a long time, the cost structure of data centers has gradually shifted from a single hardware investment issue to a comprehensive economic problem intertwined with resource scheduling, energy efficiency control, and lifecycle management. The differences in computing power density, power consumption curves, and task types among heterogeneous chips make the traditional configuration approach centered on hardware performance difficult to achieve stable cost control targets. Based on the evolution characteristics of computing power deployment and operational data features, this paper starts from the relationship between the cost composition mechanism and scheduling behavior, and constructs an optimization path framework centered on the task mapping mechanism, energy efficiency evaluation rules, and the cost feedback loop. It reveals the intrinsic relationship between resource allocation and economy in multi-generation collaborative architectures. This path emphasizes dynamic adjustment based on the operating status, making the computing power deployment strategy have both performance orientation and cost constraint attributes, providing a sustainable economic management idea for data center operations in high-density computing environments.

## Keywords

Multi-generation collaborative architecture; CPU-GPU heterogeneous system; Data center cost; Resource scheduling optimization

## Introduction

In the dynamic process of continuous expansion of high-performance computing and artificial intelligence tasks, data centers have gradually formed a heterogeneous structure where multi-generation CPUs and GPUs coexist. Although this structure enhances computing power elasticity, it also amplifies energy consumption differences and resource mismatch risks. The differentiation of different generations of chips in terms of power density, load adaptability, and depreciation cycle has transformed the cost issue from being limited to the procurement level to a systematic issue throughout the scheduling behavior and operational strategies. If the configuration logic still follows the peak performance-oriented approach, it often leads to inefficient occupation of high-performance resources and overall economic imbalance. In the context of the continuous expansion of computing power scale, how to embed the hardware collaboration mechanism and cost control goals into the same operational framework has become a real challenge for data center operation decisions.

# 1. Deployment Forms and Evolution Mechanisms of Computing Power under Multi-generation CPU-GPU Architectures

## 1.1 Evolution of Deployment Forms under the Trend of Heterogeneous Architecture Integration

The rapid growth of high-density computing tasks has prompted data centers to gradually build heterogeneous computing power systems composed of CPUs and GPUs [1]. In scenarios where AI model inference, training, and general computing loads coexist, a unified chip deployment architecture is unable to simultaneously meet the requirements of high concurrency and high throughput. The asymmetry of chip update cycles and the interleaved procurement and iteration rhythms have led data centers to generally be in an evolving state of "coexistence of multiple generations of chips", such as different generations of GPUs like A100 and H100 jointly undertaking inference and training tasks, and different versions of Intel Xeon series being used in parallel in IO scheduling. Although the trend of architecture integration has enhanced the overall scheduling flexibility of hardware resources, it has also introduced multiple compatibility constraints at the chip level, storage level, and interconnection level. The power step differences, delay sensitivity distribution variations among multiple generations of chips, and inconsistent computing-memory-communication paths between chips make it difficult for the scheduling system to establish a unified allocation strategy when facing diversified loads and tight timing.

## 1.2 The Tension Impact of Coexistence of Multiple Generations of Hardware on Resource Scheduling

The collaborative parallel deployment of multi-generation CPUs and GPUs breaks the traditional "single-layer performance-driven scheduling" logic, forcing the resource management system to dynamically match tasks with computing power units in a complex adaptation network. Taking GPUs as an example, under the same task load, the power consumption curves, memory access delays, and core utilization rates of A100 and H100 show significant deviations. If the scheduling system continues to use the allocation strategy dominated by peak computing capacity, it is prone to cause "redundant inefficient invocation" of resources - that is, high-performance chips undertake light-load tasks, resulting in high unit computing power costs and frequent resource idling. At the same time, some mid-generation chips are limited by memory bandwidth or cache architecture, and encounter IO bottlenecks when facing new-generation high-load concurrent tasks, causing unstable response times at the scheduling nodes and overall performance degradation. A more complex situation is that the scheduling algorithm is difficult to precisely link and match the lifespan stage of the chips, energy consumption boundaries, and task time windows, thus forming a structural resource imbalance [2].

# 2. Analysis of Cost Composition and Fluctuation Factors in Data Centers under Heterogeneous Hardware Environments

## 2.1 Multi-level Cost Composition Model and Evolution of Evaluation Structure

In architectures where multiple generations of CPUs and GPUs coexist, the Total Cost of Ownership (TCO) for data centers has shifted from a traditional single structure centered on hardware procurement to a complex composite cost structure involving factors such as hardware depreciation, power consumption, cooling expenses, load scheduling, and resource waste [3]. High-generation chips, due to their high power density and large heat output, cause a non-linear increase in the pressure on power supply and cooling systems during operation, thereby raising the energy cost per unit time. Old-generation chips, although having lower energy consumption, show a significant decline in the marginal contribution of computing power, resulting in a lower output per unit of power consumption, and further increasing the cost of resource scheduling redundancy. At the same time, the fragmentation of load during equipment scheduling and resource integration processes also constitutes resource wastage at the scheduling level. During the operation period of heterogeneous architectures, the hardware lifecycle, depreciation residual value, maintainability parameters, and availability status form a complex value change chain, making it difficult to achieve dynamic matching of maintenance and replacement plans in terms of cost accuracy. This evolution of the evaluation structure directly lays the intrinsic foundation for the trend of cost uncontrollability.

## 2.2 Identification of Key Cost-sensitive Factors in Multi-generation Collaborative Architecture

The fundamental cause of cost fluctuations in heterogeneous environments does not lie in the price of a single piece of hardware or labor costs, but stems from the amplifying effect of multiple marginal factors [4]. Firstly, the impact of the chip's lifecycle stage is significant. In the early deployment stage of high-generation chips, due to insufficient

driver adaptation and task orchestration, the actual utilization rate is low, which increases the unit energy consumption and unit depreciation value. However, in the later stage, there is a problem of energy efficiency degradation, making the equipment that originally had a cost advantage become a burden for the system. Secondly, the dynamic increase in the mismatch rate of scheduling leads to a rise in resource overflow or performance deficiency imbalance, triggering abnormal energy consumption or operational bottlenecks. Additionally, the instability of the cooling system's response to peak load also constitutes a significant cost-sensitive point. When multiple high-power chips are concurrently running, the asymmetric cooling paths and environmental temperature control differences cause the system's air conditioning and cooling facilities to enter an overloaded operation range, resulting in a short-term expenditure increase far exceeding the average value, forming a high-risk section with a fluctuation amplification effect.

### **3. Cost optimization of the Data Center Based on Collaborative Architecture and Key Path Design**

#### **3.1 Establish a Load Mapping Mechanism to Enhance Scheduling Compatibility**

To avoid scheduling mismatch issues between multiple generations of chips, a resource adaptation mapping system based on task attributes should be constructed. Task types should be divided into low-latency response type, high-throughput computing type, memory-intensive model type, and mixed load group types, and different generations of GPU or CPU resources should be matched respectively. The system needs to set up a pre-scheduling evaluation module, extract key parameters such as instruction structure, bandwidth requirements, and peak computing time during the task submission process, and dynamically match them with the chip adaptation database. The mapping table should set priority calling intervals according to chip generations and allow running logs to provide feedback for adjusting the adaptation coefficient to avoid static original labels. During the scheduling process, a three-level matching mechanism of "compatibility priority - performance trade-off - redundancy protection" should be introduced. When a task fails to hit the ideal resources in the first round, it should automatically be assigned to the chip with the performance closest to the target interval, and a fallback task pool should be established to wait for the resource release window [5].

The scheduling system needs to embed a "task-resource mapping correction mechanism", continuously monitor the actual utilization rate and delay performance of resources during the operation stage, and regularly write deviation information into the mapping scheduling model for parameter fine-tuning. To avoid rigid rules, an adaptive scheduling channel should be set up. The system should conduct backtracking analysis on consecutive failed matches, extract common mismatch factors, and update the mapping matrix weights. At the deployment strategy level, the scheduling controller should be linked with the resource status monitoring platform, making the current chip's temperature, voltage, and occupancy rate identifiable indicators for the scheduling node, forming visual constraints for task allocation [6].

#### **3.2 Set Energy Efficiency Evaluation Rules and Optimize Resource Allocation Structure**

Integrate an energy efficiency evaluation mechanism into the resource allocation process to prevent the scheduling system from showing a one-sided "performance priority" tendency when allocating resources among multiple generations of chips. The core indicators for the energy efficiency scoring model should be PPW (Performance per Watt) and TPP (Throughput per Power). A chip-task combination energy efficiency scoring model should be constructed. The scheduler needs to configure a "running state power consumption collection sub-module" to record the power fluctuation curves of different chips under different task types in real time, and send this data to the energy efficiency database to form a historical energy consumption map across task dimensions [7]. The scoring model should dynamically correlate variables such as chip generations, runtime temperatures, power supply loads, and single-task response times, and establish a "resource health + running efficiency" dual-weight evaluation matrix. This will ensure that the scheduling priority is no longer dependent on static computing power labels, but has the ability to identify and rank energy consumption performance.

During the resource allocation process, the scheduling engine should first set the target energy efficiency range for the tasks to be executed, and then call the scoring model to select the combination with the best score from the available resource pool. If the resource scores are concentrated in the boundary range, the backup allocation logic will be triggered, and the scheduler will call resources based on the optimal combination and adjust the chip activation priority according to the "hot and cold level table" to avoid local overheating caused by consecutive scheduling. To ensure the long-term convergence trend of the energy efficiency curve, the "redundant core recovery mechanism"

should be executed regularly. Low-scoring chips will be converted to standby status, and their power consumption performance will be included in the resource scoring factor of the next cycle. The allocation logic also needs to consider the peak power limit of the entire machine and the load balance requirements of the power supply structure, so that energy efficiency optimization not only has local optimal attributes but is also embedded in the energy management constraints of the entire system [8].

### 3.3 Embedding Cost Feedback Loop to Achieve Adaptive Iterative Strategy

In the multi-generation collaborative architecture scheduling system, a cost feedback module based on the task level should be embedded. The energy consumption data during operation, equipment utilization duration, resource idleness ratio, task interruption frequency, and scheduling invocation logs should be constructed into a unified data feedback channel. After each task is completed, the system needs to generate a cost snapshot based on the task identifier, encode the resource consumption, system load changes, and scheduling response delay within the calculation cycle into multi-dimensional indicators, and write them into the scheduling controller log queue. The data should be called as strategy input in the next scheduling cycle to form a "task structure - scheduling path - cost performance" three-way mapping chain. To enhance data validity, the feedback system should have an outlier handling module, which marks and tracks records with power fluctuations or sudden increases in idleness, and outputs a diagnostic report for the model to eliminate.

The strategy correction process should be trained with feedback data to develop a cost prediction model [9]. This model estimates the unit cost range of each scheduling scheme under future load scenarios. The scheduler calls the prediction results before task allocation, sorts the candidate resource schemes, and combines the current resource pool status and chip health parameters to set the strategy selection threshold. The model structure should be compatible with multiple source inputs, allowing the analysis of task set performance in time windows and dynamically adjusting the scoring weights. To enhance the stability of the closed-loop operation, a strategy intervention window can be set. In cases of extreme scheduling failure rate or continuous decline in resource utilization, the system will trigger an artificial assistance correction mechanism.

## 4. Conclusion

Collaborative architecture has become the core trend of high-density computing resource scheduling. The problems of resource mismatch, energy efficiency imbalance, and strategy lag brought by it determine that cost control no longer relies on local optimization methods, but needs to build a systematic mechanism that integrates task structure, chip characteristics, and system feedback. With task mapping as the entry point, energy efficiency score as the screening basis, and operation feedback as the dynamic adjustment core, the scheduling system can break away from the dependence on static rules and turn to a low-cost evolutionary path with adaptive capabilities. For future deployment of larger-scale and more complex task graphs, the scheduling logic will further integrate cost prediction and behavior monitoring models, forming a bottom-level scheduling engine with continuously evolving capabilities in an environment with frequent resource supply-demand mismatches.

## References

- [1] Chatzistefanidis I, Nikaen N. Symbiotic agents: A novel paradigm for trustworthy AGI-driven networks. *Comput Netw.* 2025;273:111749.
- [2] Song X. Research on optimization of e-commerce supply chain logistics management model based on blockchain. *Front Soc Sci Technol.* 2025;7(3).
- [3] Dong Z, Ge X, Huang Y, et al. EG-STC: An Efficient Secure Two-Party Computation Scheme Based on Embedded GPU for Artificial Intelligence Systems. *Comput Mater Contin.* 2024;79(3):4021-44.
- [4] Nuno F. GPU acceleration of the ATLAS calorimeter clustering algorithm. *J Phys Conf Ser.* 2023;2438(1):012023.
- [5] Chiheb AA, Wael L, Faisal M, et al. Firefly algorithm and learning-based geographical task scheduling for operational cost minimization in distributed green data centers. *Neurocomputing.* 2022;490:146-62.
- [6] H R, P S. Analytical Comparison of Cloud Data Centre Services and Cost. *IOP Conf Ser Mater Sci Eng.* 2021;1022(1):012058.
- [7] Kumar M, Sharma CS, Goel S, et al. Autonomic cloud resource provisioning and scheduling using meta-heuristic algorithm. *Neural Comput Appl.* 2020;32(24):18253-71.
- [8] Tang W, Cai W, Yao Y, et al. An alternative approach for collaborative simulation execution on a CPU+GPU hybrid system. *Simulation.* 2020;96(3):347-61.
- [9] Yeganeh H, Salahi A, Pourmina AM. A Novel Cost Optimization Method for Mobile Cloud Computing by Capacity Planning of Green Data Center With Dynamic Pricing. *Can J Electr Comput Eng.* 2019;42(1):41-51.