# Electricity-Cost-Aware Scheduling of CPU-Intensive Tasks in Data Centers

Aijia Luo
*Department of Electrical Engineering*
*Tsinghua University*
Beijing, China
Aimee.aijia@gmail.com

Min Chen
*Department of Electrical Engineering*
*Tsinghua University*
Beijing, China
chenmin_ee@outlook.com

Yi Wang
*Department of Electrical Engineering*
*Tsinghua University*
Beijing, China
y-wang22@mails.tsinghua.edu.cn

Nianfeng Tian
*Department of Electrical Engineering*
*Tsinghua University*
Beijing, China
nianfengtian@126.com

Qinglai Guo*
*Department of Electrical Engineering*
*Tsinghua University*
Beijing, China
guoqinglai@tsinghua.edu.cn

***Abstract*—Modern data centers exhibit considerable potential in electricity load management, where fine-grained task scheduling is key to unlocking this potential. This paper addresses the energy efficiency optimization of CPU-intensive tasks under two-part time-of-use (ToU) electricity pricing schemes. A cost-minimizing scheduling model and optimization framework are proposed. The framework constructs a scheduling model and designs a multi-dimensional scheduling strategy for server clusters. Experiments based on real electricity price data and heterogeneous server clusters validate the effectiveness of the proposed strategy in various scenarios. Results show that, across diverse data-center settings, our approach significantly reduces energy costs; under certain electricity pricing structures and workload characteristics, savings can exceed 90%, improving responsiveness to price signals and overall efficiency. These findings provide a practical and efficient scheduling pathway for data centers participating in flexible electricity systems.**

***Index Terms*—Data center, CPU-intensive tasks, task scheduling optimization, two-part time-of-use electricity pricing, energy management**

## I. Introduction

With the rapid expansion of data centers to support artificial intelligence and cloud services, their electricity consumption has surged, placing increasing pressure on the power grid. Data centers possess inherent load regulation capabilities through mechanisms such as task scheduling and workload migration. Effectively leveraging this flexibility for optimized scheduling and application of computing tasks has become a critical research issue.

Currently, numerous studies have explored flexible scheduling optimization strategies for data centers, proposing a variety of practically applicable approaches. Geographical load balancing leverages predictive migration algorithms based on electricity price signals and the distribution of renewable energy [1], [2]. Priority-based scheduling dynamically adjusts the execution order of computing tasks according to their importance and latency tolerance [3]–[5]. Energy-aware scheduling incorporates power-aware mechanisms and energy efficiency evaluation methods to design dynamic scheduling strategies based on server power consumption states [6]. Predictive scheduling integrates forecasts of weather, workloads, and energy markets to build forward-looking optimization frameworks for data center operations [1], [6].

However, most of these treat workloads as coarse-grained units and prioritize overall resource utilization and macro-level energy efficiency, rather than prioritizing fine-grained scheduling mechanisms for CPU-intensive computing tasks. Particularly in electricity pricing response scenarios, current research seldom performs joint optimization using fine-tunable parameters such as thread count, CPU frequency, and task assignment. Additionally, models and frameworks specifically tailored to pricing mechanisms remain lacking, limiting effective responsiveness and task completion quality.

To address these challenges, this paper proposes a general-purpose scheduling framework for optimizing energy consumption of computing tasks under two-part time-of-use (ToU) electricity pricing. Specifically, we (1) formulate an optimization model that minimizes electricity cost while accounting for task deadlines, pricing fluctuations, thread count, CPU frequency, and task assignment decisions; (2) design a cluster-level scheduling strategy that integrates task scheduling principles, resource allocation logic, and pricing-aware mechanisms; (3) develop a coordinated optimization approach that jointly manages task assignment, thread control, and frequency scaling to enhance responsiveness to dynamic pricing; and (4) validate the framework's effectiveness in reducing energy cost and ensuring task completion through multi-scenario simulations.

Smart-Grid National Science and Technology Major Project under Grant 2025ZD0804800

*Corresponding author: guoqinglai@tsinghua.edu.cn

979-8-3315-8138-1

## II. Optimization Scheduling Problem

In computing task scheduling, electricity cost optimization fundamentally depends on the coupling between task execution behavior and the electricity pricing structure. At the server cluster level, this coupling manifests as the ability of scheduling strategies to match execution timing with electricity price intervals and server power states. To analyze this relationship, this section starts with the electricity pricing model and incorporates server power characteristics to define the optimization objective and formulate the electricity cost computation applicable to multi-task, multi-server environments.

### A. Objective Function

For scheduling CPU-intensive computing tasks on CPU clusters, the core objective is to minimize the total electricity cost over the scheduling period. Under a ToU electricity pricing scheme, the total cost consists of a basic capacity charge and an energy charge, denoted as:

$$C_{\text{total}} = C_{\text{basic}} + C_{\text{energy}}. \tag{1}$$

Assuming transformer capacity billing is used for the basic charge, and that it remains constant over the scheduling period, the optimization objective can be simplified to minimizing only the energy charge:

$$\min C_{\text{energy}} = \sum_{t=t_0}^{T-\Delta t} P_{\text{avg}}(t)\,\pi(t)\,\Delta t. \tag{2}$$

Here, $t_0$ is the start time, $T$ is the total scheduling duration, $\Delta t$ is the time interval granularity, $P_{\text{avg}}(t)$ is the average total power of the server cluster at time $t$, and $\pi(t)$ is the average electricity price during $\Delta t$.

Assume there are $N$ servers in the cluster. Then the total average power is the sum of each server's average power:

$$P_{\text{avg}}(t) = \sum_{i=1}^{N} P_i(t) \tag{3}$$

where $P_i(t)$ is the average power of server $i$ at time $t$.

Server power consumption is closely related to its running tasks, CPU frequency, and thread count [7]. Assuming task-induced incremental power is additive and fixed during execution, the power of server $i$ at time $t$ can be expressed as:

$$P_i(t) = \sum_{j=1}^{J} k_{ij}(t)\cdot c_{ij}(f_i(t), n_{ij}(t)) + P_{\text{idle},i}(f_i(t)) \tag{4}$$

where:

- $J$ is the number of task types,
- $k_{ij}(t) \in \mathbb{N}$ denotes the number of copies of task $j$ running on server $i$ at time $t$,
- $f_i(t)$ is the CPU frequency of server $i$ at time $t$,
- $n_{ij}(t)$ is the number of threads used for task $j$ on server $i$ at time $t$,
- $c_{ij}(\cdot)$ is the incremental power function for task $j$ on server $i$ [7],
- $P_{\text{idle},i}(f_i(t))$ represents the idle power consumption of the $i$-th server when the CPU frequency is set to $f_i(t)$.

Thus, the final objective function for the optimization problem becomes:

$$\min \sum_{t=t_0}^{T-\Delta t}\left[\sum_{i=1}^{N}\left(\sum_{j=1}^{J} k_{ij}(t)\,c_{ij}\big(f_i(t), n_{ij}(t)\big) + P_{\text{idle},i}\big(f_i(t)\big)\right)\right]\pi(t)\,\Delta t. \tag{5}$$

### B. Constraints

The optimization must satisfy several constraints:

- **Time Constraints:** All tasks must be completed within their respective deadlines.
- **Resource Constraints:** Each server has a maximum number of threads (i.e., physical CPU cores), and CPU frequencies are limited to discrete values. Some servers may limit the number of concurrent tasks of specific types.
- **Operational Constraints:** Servers must remain powered on during the entire scheduling period. Thread count and frequency must remain constant during task execution.

### C. Decision Variables

The decision variables in this optimization problem are:

- Start time for each task.
- Task-to-server assignment.
- Number of threads allocated to each task.
- CPU frequency setting of each server at each time step.

## III. Scheduling Strategy for CPU-Intensive Tasks in Server Clusters

As computing task complexity grows and energy constraints become stricter, scheduling strategies must balance performance and energy efficiency. Server clusters, as the backbone of large-scale computing, benefit significantly from optimized resource allocation in terms of reduced energy consumption and improved responsiveness. Considering variations in computing performance and power characteristics across servers, scheduling strategies should align task features with suitable resources to achieve optimal overall energy efficiency. This section presents a scheduling strategy based on fundamental principles, resource allocation logic, and electricity price responsiveness.

### A. Task Scheduling Principles

Based on the relationships between execution time, power consumption, and thread count under fixed CPU frequency [7], the energy consumption required to complete a task is a function of the number of threads used:

$$Q(n) = \frac{a_1 b_0}{n} + a_0 b_1 \cdot n + a_0 b_0 + a_1 b_1 \tag{6}$$

979-8-3315-8138-1

Here:

- $Q$ is the energy consumption,
- $n$ is the number of threads,
- $a_0$, $a_1$, $b_0$ and $b_1$ are positive constants related to task characteristics and server hardware, where $a_0 = T_0(1-q)$, $a_1 = T_0 q$,$b_0 = P_{\text{base}}$, and $b_1 = \Delta P$. [7]

To minimize energy consumption for a given task on a specific server at a fixed CPU frequency, one should choose an optimal thread count $n^*$ that minimizes $Q$. Differentiating $Q$ with respect to $n$ and setting the derivative to zero yields the theoretical optimal thread count:

$$n^* = \sqrt{\frac{a_1 b_0}{a_0 b_1}} \tag{7}$$

Since thread counts must be integers, the optimal integer thread count is:

$$n_{\text{opt}} = \arg \min_{n \in \{\lfloor n^* \rfloor,\ \lceil n^* \rceil\}} Q(n). \tag{8}$$

The optimal number of threads is positively correlated with $a_1$ and $b_0$ and negatively correlated with $a_0$ and $b_1$, indicating that the optimal thread count increases with greater task parallelizability and higher baseline server power, and decreases with higher per-thread incremental power or greater fixed non-parallelizable time. Thus, the optimal thread count depends on both task characteristics and server attributes. An example is shown as Fig. 1.

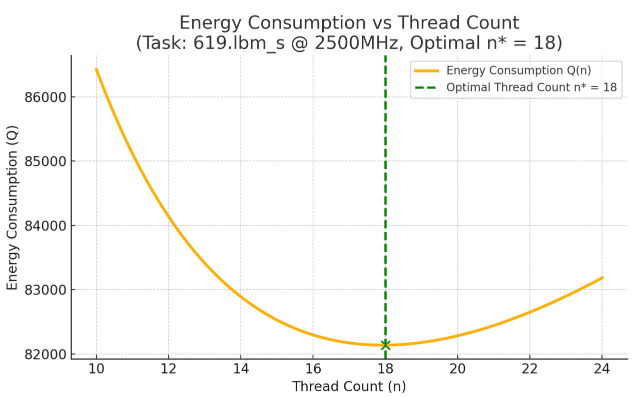
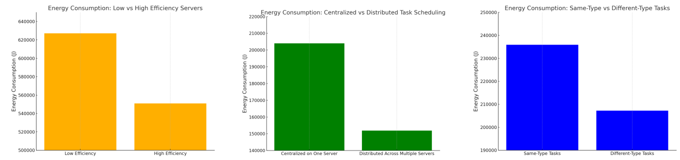



Fig. 1. Energy consumption vs. thread count

## B. Resource Allocation Logic

The results presented in [7] illustrate the impact of different allocations of computing resources on server power consumption, execution time, and energy consumption for CPU-intensive computational tasks. Based on this, the following guidelines are used for resource allocation:

- **Choose servers with higher performance-per-watt ratios** to minimize energy use for task execution.
- **Distribute tasks across multiple servers** to enable concurrent execution, avoiding bottlenecks and improving throughput.



(a) Low- vs. high-efficiency servers (b) Centralized vs. distributed scheduling (c) Same-type vs. different-type tasks

Fig. 2. Energy consumption under different task-allocation strategies.

- **Prefer heterogeneous task combinations on a single server**, such as pairing integer and floating-point tasks, to better utilize different hardware units and reduce intra-task resource contention.

The energy-saving effects of the above guidelines are illustrated in Fig. 2.

## C. Handling Time-of-Use Price Variations

To respond to electricity price fluctuations, the strategy prefers scheduling tasks during off-peak or valley pricing periods (e.g., nighttime), thereby substantially reducing operating costs. The scheduling mechanism continuously monitors price intervals and dynamically adjusts task allocation to shift workloads into lower-priced periods where possible.

# IV. Case Study

All simulation experiments in this study are based on the 1.5x agent-based electricity pricing scheme published by State Grid Zhejiang Electric Power Co., Ltd. [8], assuming a voltage level of 10 kV.

The CPU-intensive computing tasks used in the experiments are selected from the SPEC CPU® 2017 benchmark suite [9], which includes 19 integer and floating-point tasks.

The server cluster consists of two Hygon 7185 servers and two Hygon 7285H servers, for a total of four servers.

All tasks are required to be completed within a 24-hour period, starting at 8:30 AM on the first day and ending at 8:30 AM the following day.

The case study evaluates the effectiveness of the proposed energy-aware scheduling strategy under two scenarios: small-scale and large-scale task scheduling.

## A. Simulation Analysis for Small-Scale CPU-Intensive Tasks

This case involves scheduling 19 unique computing tasks, each executed once within 24 hours and subject to individual deadlines.

Figure 3 compares the energy charges before and after scheduling optimization.

The results show that after optimization, the additional energy charge attributable to task execution is reduced by 92.90%. Moreover, the optimized total electricity cost is only 8.62% of the pre-optimization total (including idle), clearly demonstrating the effectiveness of the scheduling strategy.

979-8-3315-8138-1

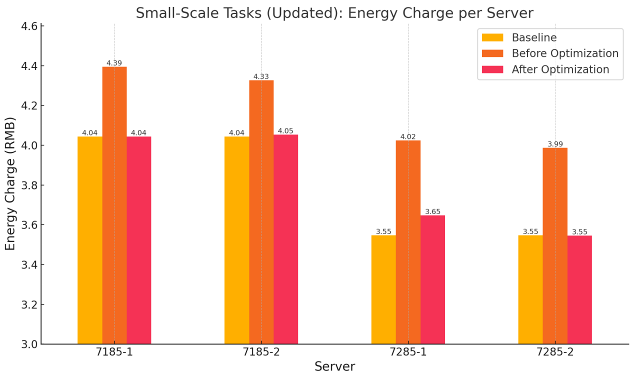



Fig. 3. Comparison of energy charges per server before and after optimization (small-scale tasks).

### B. Simulation Analysis for Large-Scale CPU-Intensive Tasks

In this case, the same 19 computing tasks are executed multiple times within their specified 24-hour windows.

Figure 4 presents the comparison of energy charges before and after optimization.

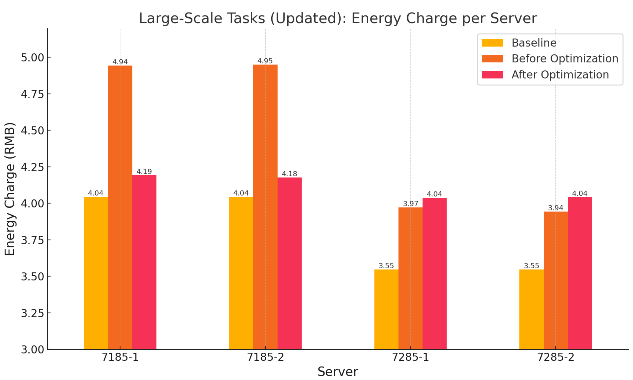



Fig. 4. Energy charge per server before and after optimization for large-scale tasks.

Again, the optimized scheduling strategy results in a 51.72% reduction in extra energy cost, confirming its practical value in large-scale computing scenarios.

## V. Conclusion and Future Work

This paper proposes a scheduling model and strategy for optimizing electricity cost in CPU-intensive computing tasks. The major conclusions are as follows:

- An optimization model is developed to minimize electricity costs by considering task deadlines, electricity price variations, CPU frequency, and thread count as fine-grained parameters.
- Coordinated control of thread count and CPU frequency is essential for improving energy efficiency and electricity price responsiveness. Selecting optimal thread counts significantly reduces task-level energy consumption. The optimal value depends on task parallelizability and server power characteristics.
- In multi-server clusters, considering performance-per-watt ratios and task heterogeneity when allocating resources can significantly enhance overall resource utilization and task completion rates.
- The proposed scheduling strategy effectively reduces energy charges associated with task execution. In representative scenarios, cost savings can exceed 90%, demonstrating its practical applicability.

Future work will expand the model's applicability to multi-period, cross-day task scheduling and dynamic task flow scenarios. Moreover, we will explore the model's portability and robustness in GPU clusters and heterogeneous computing platforms.

## References

[1] Flex-power.energy, "How Flexible Can Data Centers Be in Their Electricity Consumption?," [Online]. Available: https://flex-power.energy/energyblog/data-centers-electricity-consumption-flexibility/, 2023.

[2] Z. Liu, H. Zhu, and Z. Liu, "Data Center Demand Response: A Survey," *ACM Computing Surveys (CSUR)*, vol. 55, no. 1, pp. 1–39, 2022.

[3] H. Wang, J. Gao, J. Huang, J. Yu, S. Qian, and H. Yao, "Power system operational reliability evaluation method considering data center load flexibility," *Automation of Electric Power Systems*, 2023. [Online]. Available: https://epjournal.csee.org.cn/dlxtbhykz/cn/article/pdf/preview/10.19783/j.cnki.pspc.230348.pdf

[4] Y. Wang, Y. Yuan, Y. Xue, and Y. Cao, "Collaborative Optimization of Data Centers and Power Systems under the Background of Energy Internet (Part II): Opportunities and Challenges," *Proceedings of the Chinese Society for Electrical Engineering*, vol. 41, no. 23, pp. 8074–8088, 2021. [Online]. Available: https://www.researchgate.net/publication/357240003_nengyuanhulianwangbeijingxiashujuzhongxinyudianlixitongxietongyouhuaerjiyuyutiaozhan

[5] J. Wang, B. Zhou, F. Zhang, X. Shi, N. Zeng, and Z. Liu, "Survey of data center energy consumption models and energy efficiency algorithms," *Journal of Computer Research and Development*, 2019. [Online]. Available: https://crad.ict.ac.cn/fileJSJYJYFZ/journal/article/jsjyjyfz/HTML/2019-8-1587.shtml

[6] A. El-Gharabawy, M. A. El-Dali, and Y. M. El-Saadany, "Flexibility-Based Energy and Demand Management in Data Centers," *Energies*, vol. 12, no. 17, p. 3301, 2019.

[7] A. Luo, M. Chen, Y. Wang, N. Tian, and Q. Guo, "Load flexibility modeling of CPU-intensive computing tasks in data centers," TechRxiv, preprint, Oct. 9, 2025. [Online]. Available: https://doi.org/10.36227/techrxiv.176003095.52850256/v1. Accessed: Oct. 11, 2025.

[8] State Grid Zhejiang Electric Power Co., "1.5 Times Proxy Power Purchase Price Table" [Online]. 2024 [Accessed: Nov. 28, 2024]. Available: https://www.sx.gov.cn/art/2024/11/28/art_1229366601_4199797.html

[9] Standard Performance Evaluation Corporation, *SPEC CPU® 2017 Benchmark Suite* [Online]. 2017 [Accessed: Jun. 6, 2025]. Available: https://www.spec.org/cpu2017

979-8-3315-8138-1