北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

PortEdge: AI-Driven Sustainable Workload Coordination for Smart Port Operations

ADOLF K.Y. NG
Dean and Chair Professor, Faculty of Business and Management
Director, International Centre for Resilient Supply Chains
BNU-HKBU United International College
adolfng@uic.edu.cn

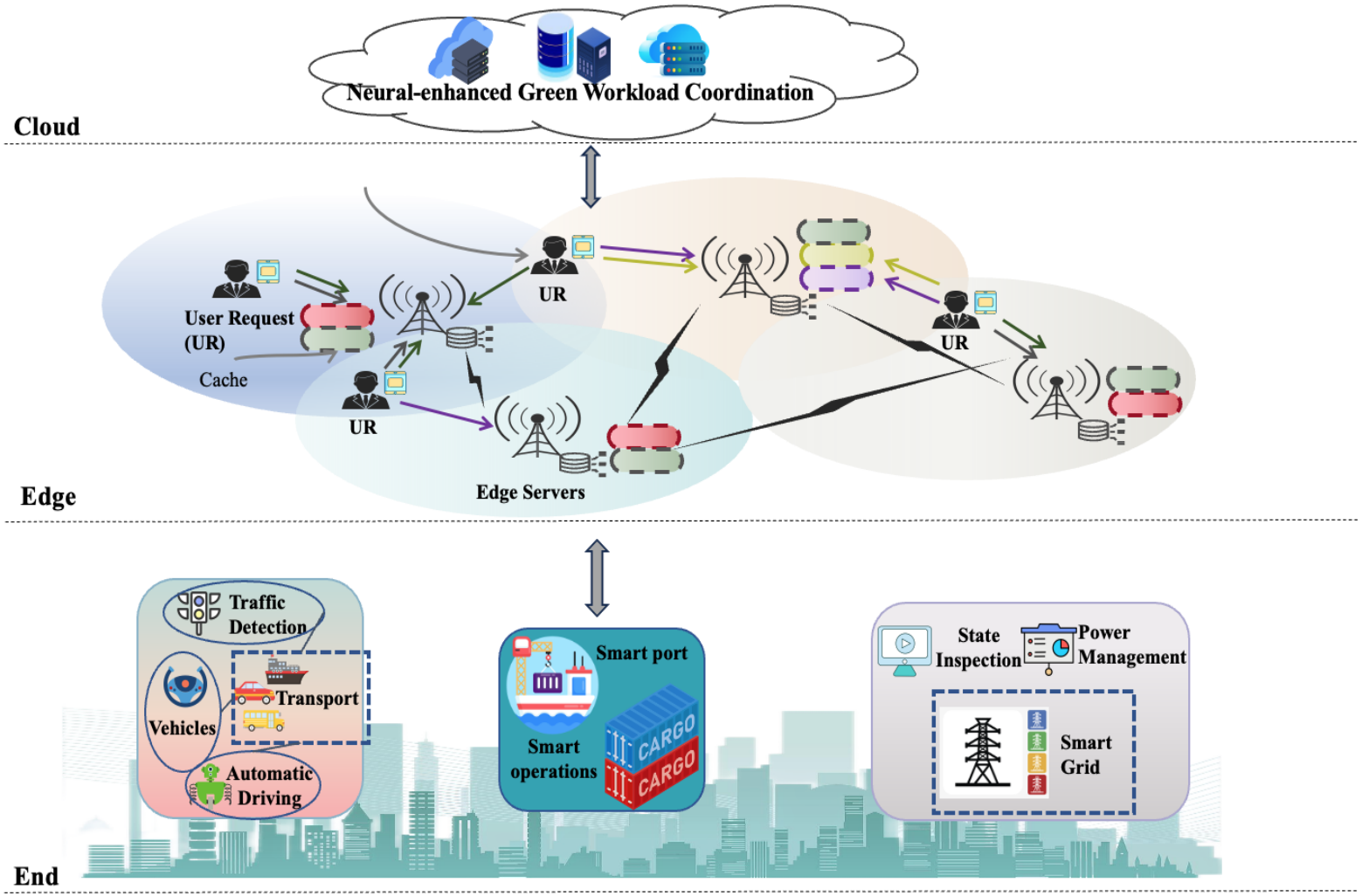
PortEdge: The AI Revolution in Smart Port Operations

北京师范大学香港浸会大学 联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Motivations:

- Ports face increasingly complex operational demands, including diverse shipping requirements, varied cargo types, and fluctuating port traffic..
- Efficient workload coordination and resource scheduling are crucial for deploying smart port operations across a network of interconnected port facilities.
- In resource-constrained port environments, the management of operations, from cargo handling to ship scheduling, often exceeds the capacity of individual systems.
- Leveraging AI-driven strategies to allocate tasks across different port nodes and operations, known as " model parallel " modes in computing, is necessary to optimize performance.

Overview of End-edge-cloud Networks in Port Operations:



Question: How can smart port operations be effectively managed and optimized?

➔ We need to resolve the workflow first.

How to Handling PortEdge Applications

北京师范大学 香港浸会大学 联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

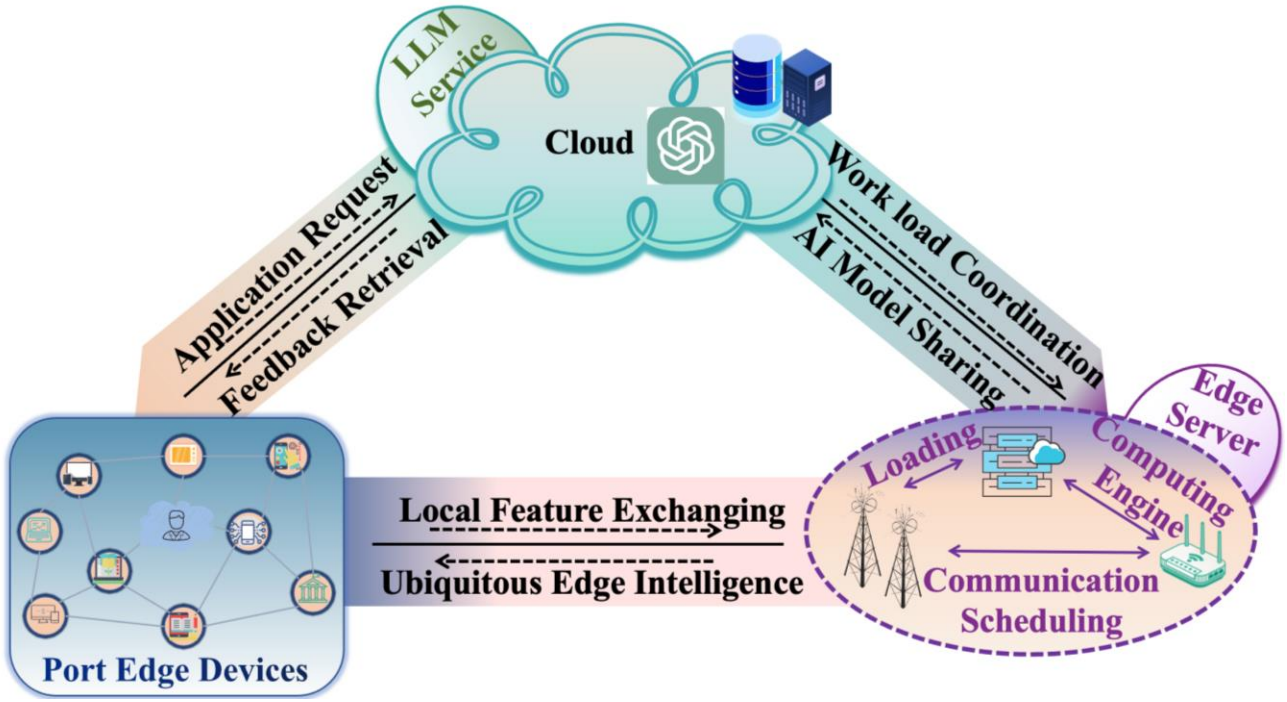
Workflow of Handling UEI Applications:

- The precondition for handling UEI applications is to translate the human instructions to job commands via in-cloud large language models (LLMs).
- The latency mainly comes from (1) uploading the input samples or latent features through the network and (2) understanding the task definition by using LLMs.
- The former measures the transmission efficiency that the cloud server can receive the data source within a reasonable deadline, and the latter reflects the analysis speed to quickly understand task objectives.
- Together with these two terms, we can get the global latency to respond to human instructions and resolve them into a series of job commands.

Summary:

- We treat these two terms as constraints to formulate the major optimization problem and define the system model.

Overview: we need to coordinate the workload among cloud, edge server and port end devices.



System Model

北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Key Points of System Model:

- We need to divide the model following the order of layers and allocate the neurons to different devices.
- A neuron corresponds to a specific function in the computation graph marked in different colors.
- Assigning a neuron to different devices will yield different performance of job completion time and energy cost.

Key Operations:

- Model Partition
- Function Assignment

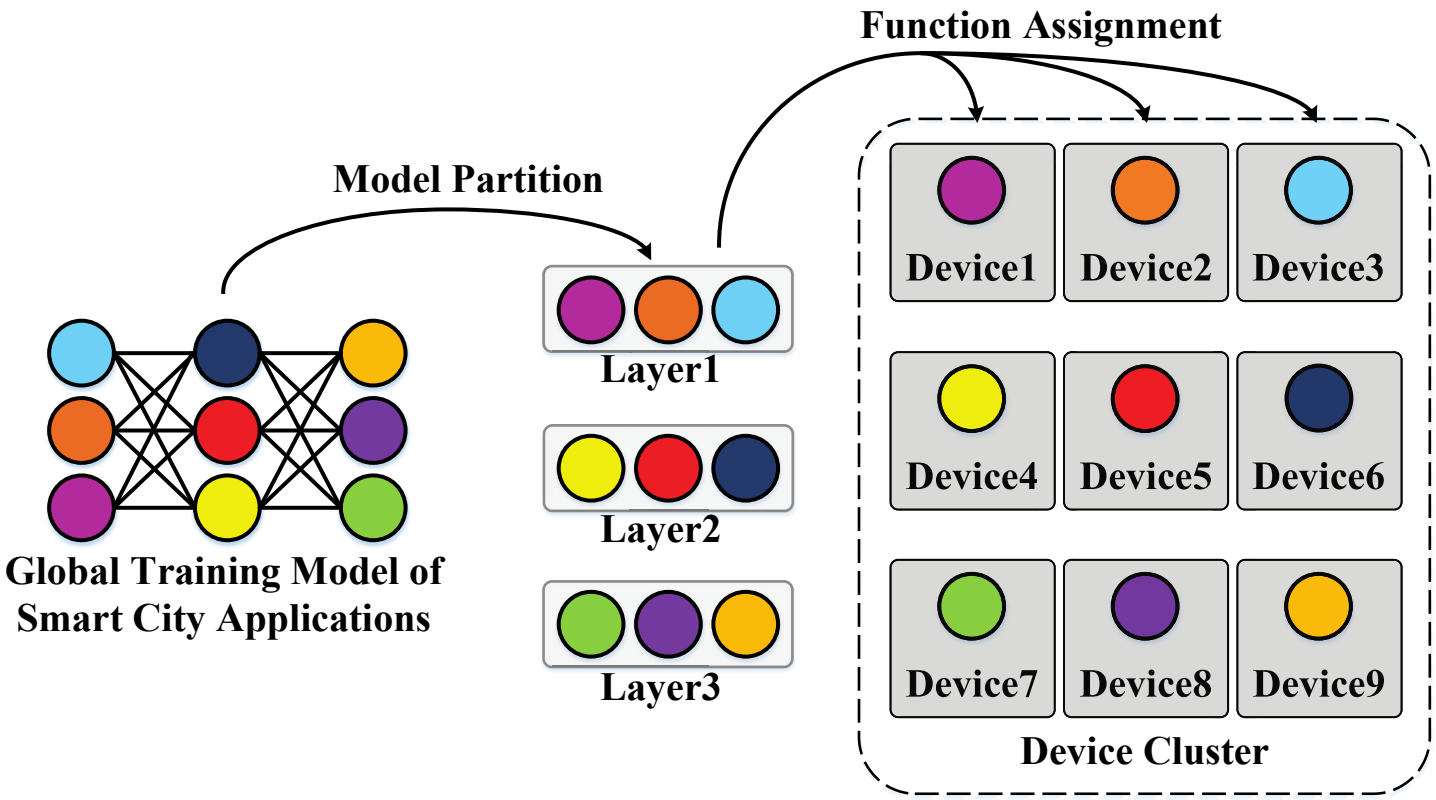
Objectives:

- minimizing job completion time
- Reducing runtime energy cost

Our Solution:

- Green Workload Coordination ➔ **PortEdge**

Visualization of Model Partition and Function Assignment:



Definition of Job Completion Time

1. As finishing the function of the subtask allocated to device i requires going through the entire local data D_i, we have the function completion time as:

$$t_{i,comp} = \sum_{j=1}^{|D_i|} n_{i,j} \cdot \frac{\delta_i}{f_i \cdot u_i}, \quad i \in \mathbb{M}, \quad d_{i,j} \in D_i$$

2. After the computation stage is done, we need to transmit the output data from device i to the devices belonging to the next layer. The communication time can be described as:：

$$t_{i,comm} = \frac{o_i}{B_i}, \quad i \in \mathbb{M},$$

3. Combining the two stages of computation and communication, the per-iteration time on device i is described as:

$$t_i = t_{i,comp} + t_{i,comm}, \quad i \in \mathbb{M}$$

4. the layer completion time is bounded by the slowest device handling the corresponding function, which can be described as

$$t_l = \max_{i \in V_l}(t_i), \quad V_l \in \mathbb{V}$$

5. By marking the layer completion time t_l with current iteration index k, the total job completion time to conduct the training workflow is expressed as:

$$T = \sum_{k=1}^{K} \sum_{l=1}^{|\mathbb{V}|} t_{l,k},$$

Definition of Runtime Energy Cost:

1. Based on the classical energy cost model[1] of portable devices, the computational energy cost to finish the processing of D_i can be described as

$$E_{i,comp} = \eta_i \cdot \sum_{j=1}^{|D_i|} n_{i,j} \cdot \delta_i \cdot (f_i \cdot u_i)^2,$$

2. the energy cost due to the heat dissipation of workload computing and network communication on device i can be calculated as:

$$E_{i,heat} = \int_{t_0}^{t_0+t_{i,comp}} g(f_i^2)dt + \int_{t_0+t_{i,comp}}^{t_0+t_{i,comp}+t_{i,comm}} h(f_i^2)dt,$$

3. Combining these two perspectives of energy cost, we can describe the energy cost on device i in current iteration k as:

$$E_{i,k} = E_{i,k,comp} + E_{i,k,comm}$$

4. the total energy cost to finish the job is calculated as:

$$E = \sum_{i=1}^{|\mathbb{M}|} \sum_{k=1}^{K} E_{i,k}\ , \quad i \in \mathbb{M}$$

[1] T. D. Burd and R. W. Brodersen, “Processor design for portable systems,” VLSI Signal Processing, vol. 13, no. 2-3, pp. 203–221, 1996.

Problem Formulation and Optimization Target

北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Analysis:

1. With the description of job completion time and energy cost, we can transfer our target into the optimization problem to minimize these two factors：

2. We use the non-negative hyper-parameter α $(0 \leq \alpha \leq 1)$ to adjust the priority weights between energy cost and job completion time.

3. Difficulties in solving through traditional optimization methods:

- There are numerous dynamic attributes, such as the current CPU remaining computing power percentage u, CPU clock frequency f, and available network bandwidth B.
- The model loss function $L_k(\omega)$ presents nonlinear-constrained conditions.
- The system can only obtain the current resources and task states, lacking complete prior knowledge of the entire training process.
- The partitioning of datasets and models directly impacts the execution time and energy consumption of tasks, resulting in a huge solution space when relying on optimization methods to solve it.

Formulation:

$$\min : \alpha E + (1-\alpha)T,$$

$$\mathcal{L}_k(\omega) \leq \epsilon,$$

$$1 \leq k \leq K,$$

$$0 \leq u_i \leq 1,$$

$$0 \leq f_i \leq f_i^{\max},$$

$$0 \leq B_i \leq B_i^{\max},$$

$$D_i \in \mathbb{D}, \quad V_l \in \mathbb{V},$$

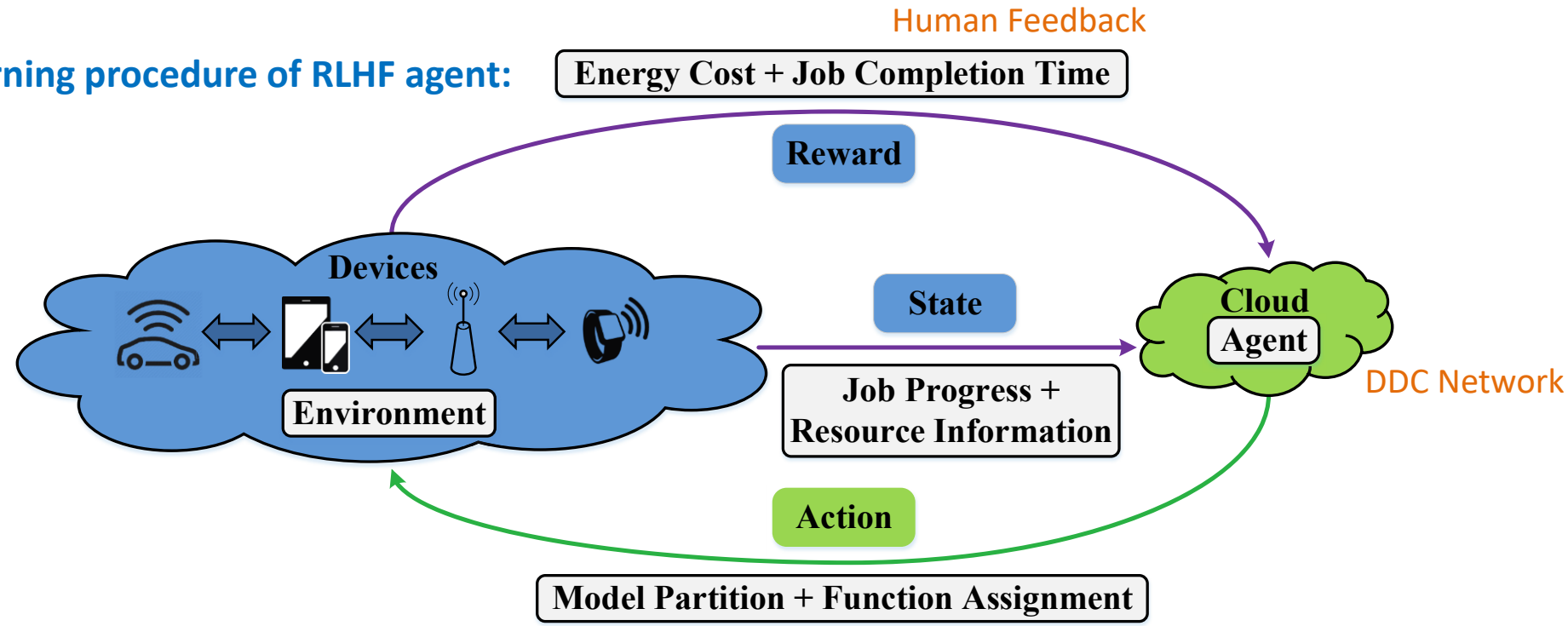
Our PortEdge Methodology:

➔ We establish an intelligent agent to solve the above problem.

PortEdge Methodology: RLHF Agent

Gist: We follow the *Reinforcement Learning with Human Feedback* (RLHF) paradigm and employ the Dynamic Discrete Choice (DDC) network to design our green workload coordination algorithm.

The iterative learning procedure of RLHF agent:



Key:

- Our RLHF agent is implemented and executed on the parameter server located on the cloud side.
- By employing an iterative process involving action, state, and reward, the RLHF agent will keep acquiring self-motivated knowledge and finally be able to optimize the strategies of green workload coordination in a long-term view.

PortEdge Methodology: Learning Procedure

北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

State Space: (1) training progress reflected by iteration index k, (2) current available network bandwidth, denoted as B_i, (3) current CPU status, denoted by frequency fi, (4) current available CPU computation capacity, and (5) loss value $L_k(\omega)$ under current model parameter ω. $s_k = \langle k; \mathcal{L}_k(\omega); u_1, u_2, \cdots, u_M; f_1, f_2, \cdots, f_M; B_1, B_2, \cdots, B_M \rangle$

Action Space: The action space A reflects the strategy of model partition and function (neuron) assignment to different devices. $a_k = \langle V_1, V_2, \cdots, V_L; D_1, D_2, \cdots, D_M \rangle$

Reward: The RLHF agent takes the reward as feedback to polish its DDC network, where the reward is directly related to to the effectiveness of the agent's action. Given an iteration with index k, the RLHF agent will analyze the status of job completion time and runtime energy cost in the entire edge-cloud network. $r_k = -\alpha E - (1-\alpha)T$

DDC Network: Consider a dataset denoted as D and each sample inside falls in the space of $\{s_h, a_h\}$. Note that D is comprised of n trajectories, which have been gathered through the observation of human decisions by the DDC model. The objective is to acquire knowledge about the optimal policy π of the underlying Markov Decision Process (MDP) process. Therefore, the policy generated by the RLHF agent $\pi_h(a|s) = \frac{\exp(\mathcal{Q}_h^\pi(s,a))}{\sum_{\hat{a}\in\mathcal{A}} \exp(\mathcal{Q}_h^\pi(s,\hat{a}))}$

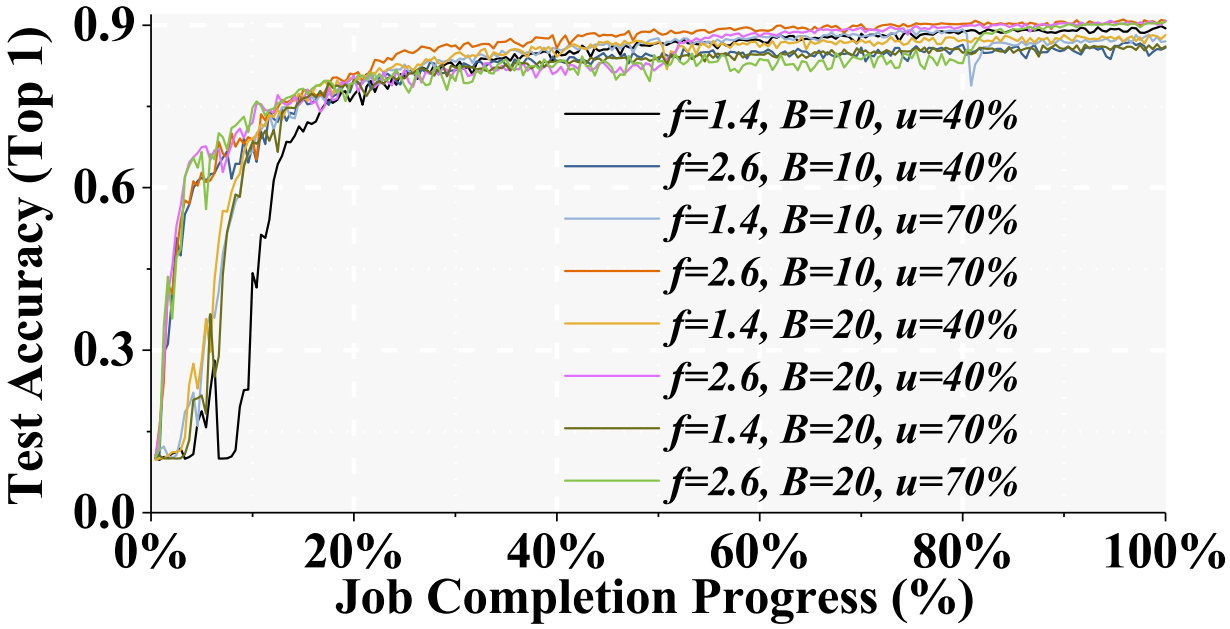
Step and State Transition: As RLHF learning procedure contains a series of steps, we mark each iteration of the distributed training job as a step. At the end of current iteration, we can re-conduct the workload placement among the devices based on agent's action and continue the training procedure for the next iteration. After that, the workflow of DRL learning goes into the next step.

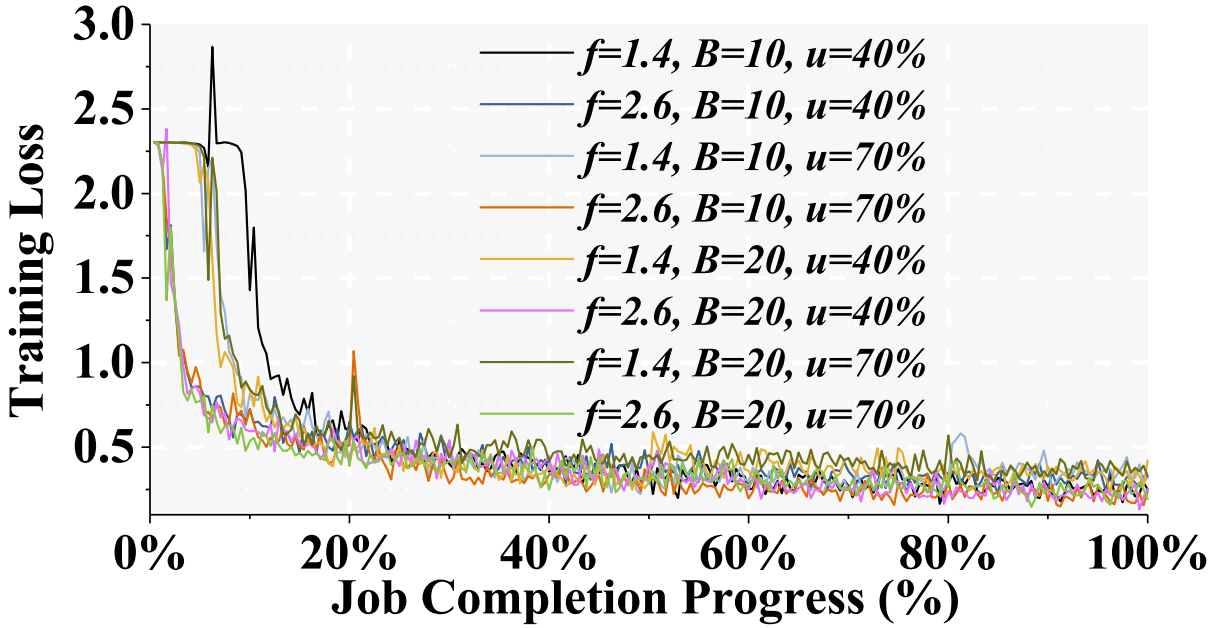
➔ **The RLHF agent learning step follows the same index of the training job iteration k.**

Performance Evaluation

北京师范大学 香港浸会大学 联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Quantitative Analysis of Convergence Efficiency:





Performance Highlights:

- The training procedure of PortEdge achieves a stable convergence and yields a high top one test accuracy.
- PortEdge can achieve better performance with a higher CPU cycle frequency and remaining computation capacity, while the available bandwidth yielding slight impact.

Summary:

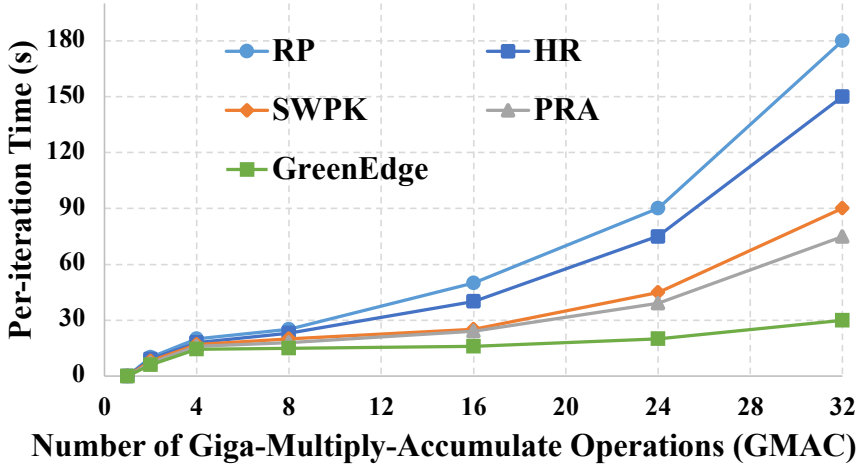
- This phenomenon verifies the significance of optimizing the model partition and function assignment among the cluster, so as to accelerate the convergence speed and reduce workload overhead.

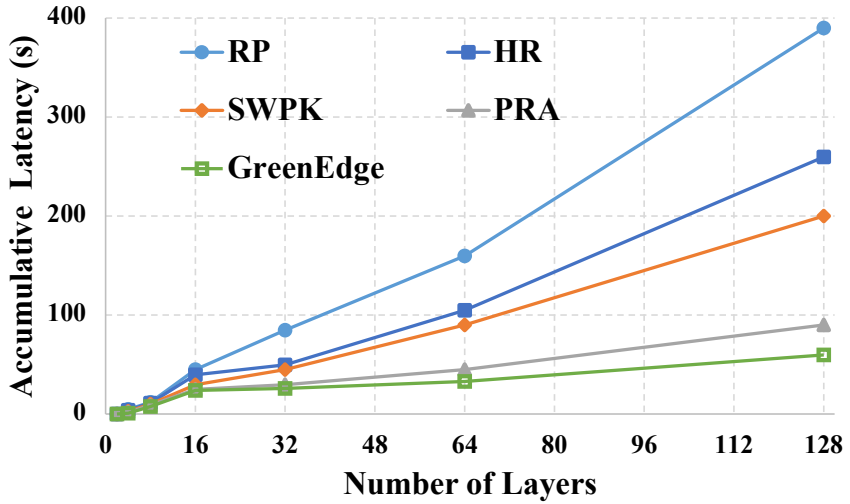
- Match edge environment: image classification on the MobileNet model using the Fashion MNIST and CIFAR-100 datasets.

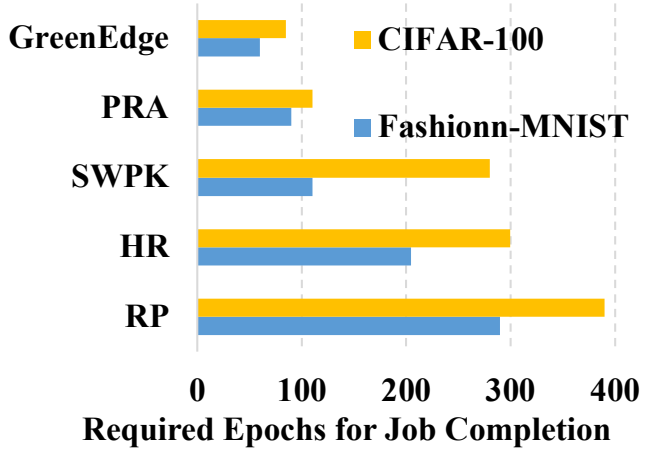
Performance Evaluation

北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Quantitative Analysis of Average Job Processing Speed:







Performance Highlights:

- PortEdge significantly reduces per-iteration time and presents a nearly linear increase to the neural network scale.
- PortEdge incoporates RLHF to enhance the framework efficiency, achieves a significant reduction in the accumulative latency caused by straggler devices.
- PortEdge requires much fewer epochs for model convergence, resulting in a notable reduction of computational overhead.

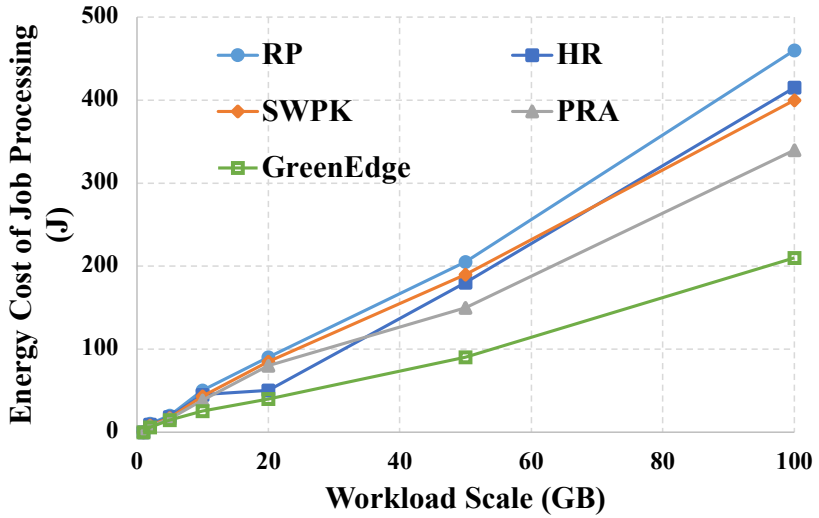
Summary:

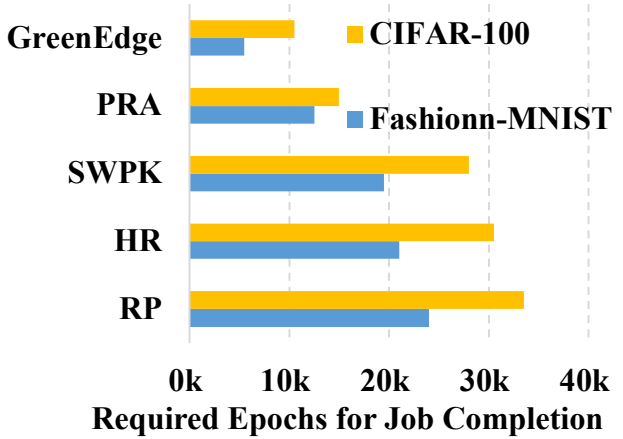
- PortEdge improves the average job processing speed by up to 1.63×, 2.28×, 3.79× and 5.13× over the previous PPO-based-resource-allocation (PRA), Static-with-prior-knowledge (SWPK) , Heuristic-rule (HR) , and Random-partition (RP) solutions, respectively.

Performance Evaluation

北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Quantitative Analysis of Runtime Energy Cost:





Performance Highlights:

- PortEdge holds a much slower growth in the curve and consumes less energy across various workload configurations.
- PortEdge successfully alleviates energy wasting by precisely placing the workload to suitable edge devices.
- PortEdge holds superior energy-saving efficiency by determining appropriate workload coordination strategies.

Summary:

- PortEdge can provide an overall energy saving improvement by up to 44.69%, 68.16%, 72.37% and 75.81% over the previous PPO-based-resource-allocation (PRA), Static-with-prior-knowledge (SWPK) , Heuristic-rule (HR) , and Random-partition (RP) solutions, respectively.

北京师范大学香港浸会大学联合国际学院
BEIJING NORMAL UNIVERSITY · HONG KONG BAPTIST UNIVERSITY
UNITED INTERNATIONAL COLLEGE

Thank you!

Contact Email:
adolfng@uic.edu.cn