Journal of Network and Computer Applications 242 (2025) 104215

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Research paper

Active RIS-assisted task partitioning and offloading for industrial edge computing☆

Mian Guo [a], Yuehong Chen [a,*], Zhiping Peng [b,c], Qirui Li [c], Keqin Li [d]

[a] Guangdong Polytechnic Normal University, China
[b] Jiangmen Polytechnic, China
[c] Guangdong University of Petrochemical Technology, China
[d] Department of Computer Science, State University of New York, New Paltz, USA

ARTICLE INFO

Keywords:
Multi-access edge computing (MEC)
Reconfigurable intelligent surface (RIS)
Partial offloading
Industrial Internet of Things (IIoT)
Delay guarantee

ABSTRACT

In Industry 5.0, smart devices in intelligent factories will generate numerous computation-intensive tasks that require low latency. Due to the limited computation resources of local devices, it is required to partition and offload tasks to edge servers via wireless networks for end-edge collaborative computing. However, intelligent factories are usually located in low-rise buildings. The direct offloading paths between smart devices and edge servers are vulnerable to being obstructed by high-rise buildings and trees, leading to intolerable long task offloading delays and even failure in offloading. To tackle this problem, we develop an active reconfigurable intelligent surface (RIS)-assisted end-edge collaborative task partitioning and offloading model, which assists task offloading by reflecting communication signals through the active RIS. We propose to maximize the system utility by jointly optimizing the task partitioning and offloading decisions, reconfiguring the phase shift and amplification factor of the active RIS, and communication and computation resource allocation, aiming at energy-efficiently providing delay guarantee to industrial computation tasks. We formulate, decompose, and theoretically analyze the problem. The upper and lower bounds of offloading decisions, transmission powers, and computation resources constrained to delay bounds have been analyzed. Based on the analytical results, a two-stage heuristic algorithm, RISADA, has been proposed to address the problem. The results demonstrate the efficiency of our proposal for the delay guarantee while reducing energy consumption.

1. Introduction

There are three key features in Industry 5.0: human-centricity, sustainability, and resiliency (Leng et al., 2022). To support a human-centric and sustainable production process, almost all smart devices (SDs), such as machines, productions, and humans wearing various sensors, will generate massive amounts of data. For maximizing the data value to support intelligent industrial applications, e.g., machine condition control, fault diagnosis, and intelligent production scheduling (Zhang et al., 2024a), these data require high-performance and low-delay computing via various artificial intelligence (AI) algorithms, such as machine learning (ML), deep learning (DL), reinforcement learning (RL), and so on (Han et al., 2024), forming a massive amount of computation tasks with various data sizes and tolerating distinct delays. Offloading computation tasks from factories to a cloud center for cloud computing would cause tasks to experience long offloading delays, e.g., hundreds of microseconds, which cannot meet the low delay requirement of delay-sensitive and computation-intensive industrial applications. Moreover, industrial data is easily exposed to public networks and computing centers in the cloud computing paradigm, raising security and privacy issues.

In recent years, multi-access edge computing (MEC) has been considered a significant computing paradigm enabling industry 5.0. MEC supports low delay and high security by deploying computation resources at a wireless network one-hop away from the data source (Akhlaqi and Mohd Hanapi, 2023). However, the wireless network and MEC computation resources are scarce and limited compared to wired networks and clouds. End-edge collaborative partial offloading is important (Peng et al., 2023; Chen et al., 2024), which partitions computation tasks into multiple subtasks and distributes them to distinct computing nodes (such as computing-capable MEC servers and Internet of

☆ This work was supported in part by the National Natural Science Foundation of China [grant number 62273109].
* Corresponding author.
E-mail addresses: mianguo@gpnu.edu.cn (M. Guo), yhchen2001@126.com (Y. Chen), pengzp@gdupt.edu.cn (Z. Peng), liqirui@gdupt.edu.cn (Q. Li), lik@newpaltz.edu (K. Li).

https://doi.org/10.1016/j.jnca.2025.104215
Received 13 November 2024; Received in revised form 8 April 2025; Accepted 6 May 2025
Available online 23 June 2025

Things (IoT) end devices) to perform distributed collaborative machine learning, such as federated learning and split learning (Lin et al., 2024). For example, with the split learning and MEC paradigm, amounts of intelligent applications, such as split learning-based convolutional neural networks (CNNs) and recurrent neural networks (RNNs), can be deployed within an industrial environment to achieve better manufacturing efficiency and product quality (Jia et al., 2024). Since the latency of a task is affected by the latest completed subtask, *how to partition a task and allocate communication and computation resources for subtasks deserve further study*.

In an industrial environment, data sources (e.g., machines, IoT devices, productions) are located inside a factory, which is generally located in low-rise buildings. High-rise buildings and trees will severely obstruct the direct communication links between the data source and the MEC server (Tan et al., 2024; Zhi et al., 2022). Accordingly, a reconfigurable intelligent surface (RIS) must be placed between the data source and the MEC server to improve task upload efficiency to meet the low delay requirements of industrial tasks. RIS, also called intelligent reflecting surface (IRS), is a planar array composed of a large number of passive elements that reflect electromagnetic signals in a desired manner, thereby reconfiguring wireless transmission properties (Shi et al., 2023). Active RIS employs power amplifiers to actively amplify the reflected signals, aiming at enhancing the wireless propagation capability at the expense of consuming extra power.

Although RIS-assisted task offloading in MEC has been studied in recent years (Tan et al., 2024; Yu et al., 2023; Lv et al., 2024; Liu et al., 2024), the joint active RIS and end-edge collaborative partial offloading in a multi-user industrial edge computing environment still faces considerable challenges. Firstly, the partitioning and the offloading ratio of a task affects not only the delay performance and energy consumption of itself but also that of others by involving the workload competing for the shared wireless and computation resources. Secondly, since the incident angles from different SDs to the same reflecting element of RIS are distinct due to their particular positions, they expect to reconfigure the RIS in various ways to maximize their own wireless performance. It is a big challenge to optimize the phase shift matrices in a multi-user RIS environment to benefit as many SDs as possible (Tan et al., 2024). Thirdly, the delay performance and energy efficiency of the RIS-assisted MEC system are simultaneously affected by multiple types of decisions distributed in distinct network elements, e.g., the task partitioning and offloading decisions in smart devices, the reconfigured phase shift and amplification factors of the active RIS, and the computation resource allocation policies in MEC servers. These types of decisions affect each other, further complicating the problem (Zeng et al., 2024; Zhou et al., 2025). In addition, since deep learning and reinforcement learning algorithms and their variants usually make one type of decision, it is challenging to collect distributed state information to make multiple types of decisions via deep learning or reinforcement learning.

Motivated by the above discussions, this paper studies an active RIS-assisted task partitioning and offloading problem in an industrial MEC environment where an active RIS is deployed to assist partial task offloading. The goal is to energy efficiently satisfy the delay requirements of computation tasks by jointly optimizing (1) task partitioning and offloading decisions in distributed SDs, e.g., how to partition computation tasks, how much ratio will be offloaded; (b) the beamforming of the active RIS, including phase shift and amplification factor; (c) computation resource allocation decisions in MEC servers for partial offloaded tasks; (d) SDs' transmission power for partial offloading. We propose a two-stage heuristic algorithm to solve the problem based on the results of the theoretical analysis.

Compared with existing heuristic and AI-based algorithms (Tan et al., 2024; Yu et al., 2023; Zhang et al., 2024b), we theoretically analyze the upper and lower bounds of offloading decisions, transmission powers, and computation resource allocation constrained to delay requirements and energy consumption, the joint decisions are based on the analytical results. Accordingly, the decisions in our proposal are more interpretable. In addition, in our proposal, the joint decisions are from multiple partial decision makers (e.g., SDs, RIS controllers, MEC servers, etc.) based on the up-to-date decisions of others and analytical results. Our algorithm could run in an edge computing environment with low computing capability. Our main contributions include:

- We formulate a joint computation offloading and active RIS optimization problem (JCORO) in an industrial MEC for the delay guarantee and energy efficiency.
- The problem is decomposed into two concatenated subproblems, task partitioning and offloading (PO), and joint active RIS optimization and MEC resource allocation (RORA). The properties of PO and RORA constrained to the delay guarantee and energy efficiency have been theoretically analyzed.
- A two-stage heuristic scheme (RISADA) has been proposed to solve JCORO based on the theoretical analysis. In particular, RISADA mainly consists of two concatenated stages; in stage A, the RORA subproblem is solved via three concatenated algorithms, which yields the optimum phase shift and amplification factor, the achievable transmission rate, and the allocated MEC computation resource for offloading subtasks; then, in stage B, we update the task partitioning and offloading decisions with the DEEPO algorithm for the PO subproblem. The two stages repeat a few times and converge to a one-shot solution for the JCORO problem.

The rest of this paper is organized as follows. The related work is discussed in Section 2. Section 3 describes the system, active RIS, task partitioning, computing, energy consumption, and utility models. Then, we formulate and decompose the problem in Section 4. The problem is theoretically analyzed in Section 5. Section 6 describes the proposed RISADA. Section 7 evaluates the performance. Finally, Section 8 concludes the paper.

2. Related work

Industrial edge computing mainly studies how to develop edge computing in industrial IoT (IIoT) for improving industrial intelligence.

Edge computing without RIS. A number of edge computing algorithms have been explored for achieving various objectives (Guo et al., 2023b; Songhorabadi et al., 2023; Peng et al., 2024). The IIoT applications are usually computation-intensive and delay-sensitive. Delay performance has been a focus of attention since the birth of edge computing. To optimize the delay performance of computation tasks in an IIoT MEC system, an RL-based offloading scheme has been studied (Deng et al., 2023). The proposal uses Q-learning to make offloading decisions while using deep deterministic policy gradient (DDPG) to optimize the system performance (Deng et al., 2023). Chen et al. released the assumptions of fixed communication time and arbitrarily splitting the workload. They proposed a computing model based on the pyramid to reduce latency in distributed edge computing (Chen et al., 2023). Energy consumption is another focus of attention in edge computing. To minimize the energy consumption of IoT devices constrained to latency requirements, Qian et al. have jointly optimized the computation offloading, nonorthogonal multiple access (NOMA) transmission, and computation resource allocation in IIoT (Qian et al., 2021). To minimize the time consumption and energy consumption of the intelligent transportation system, Zhao et al. have jointly optimized the offloading decisions, caching strategies, computation resource allocation and transmission power allocation via a multi-task multi-objective optimization algorithm (Zhao et al., 2025). Moreover, cost-aware edge computing has been studied. Dai et al. have studied the task co-offloading problem in a device-to-device (D2D) assisted MEC in IIoT. The object is to minimize system cost by making offloading decisions on where to offload the tasks (Dai et al., 2023). The joint power control and computation resource allocation problem has been transformed into a Markov decision process (MDP), and solved

via a DRL-based dynamic resource management algorithm (Chen et al., 2021). In a personalized MEC computation offloading environment, the welfare is a person's focus; thus, Su et al. proposed a truthful combinatorial auction (TCA) mechanism to maximize the social welfare in such an environment (Su et al., 2023). Considering the dynamic arrival properties of offloading requests and workloads in servers, a modified generalized second price (GSP)-based algorithm has been proposed to make pricing and resource decisions for maximizing social welfare (Habiba et al., 2024). In addition, recently, some literature has focused on deploying distributed collaborative machine learning (e.g., federated learning, split learning) in edge networks to take full advantage of MEC (Lin et al., 2024; Jia et al., 2024).

Edge computing with RIS. In recent years, more and more researchers have focused on RIS/IRS-assisted edge computing to improve wireless performance. The IRS-aided computation offloading from two users to an edge cloud over NOMA and time-division multiple-access (TDMA) has been studied for minimizing the total delay of two users (Zhou et al., 2020). A dynamic task scheduling strategy involving joint processor allocation and IRS optimization in IRS-aided vehicular networks has also been studied. The simulations illustrated the efficiency of the proposal in task offloading rate, computation rate, and finish rate (Zhu et al., 2022). Besides passive RIS-assisted MEC, joint computation offloading and transmission performance optimization in a hybrid active-passive RIS-aided MEC have also been studied (Xie et al., 2024). The joint beamforming for RISs and base stations for reconstructing transmission channels to improve system capacity has been studied in Zhang et al. (2024b). The problem of joint offloading, communication, and computation resource allocation in an IRS-assisted NOMA MEC have been explored in Yu et al. (2023). The authors designed a Lyapunov-based mixed integer deep deterministic policy gradient scheme to determine the optimum solution. To minimize the consumed energy of smart terminals (STs), Sun et al. have jointly optimized the local CPU frequencies of STs and phase of IRS (Sun et al., 2022).

Discussions. AI algorithms, such as DDPG and DRL as well as their variants, have been explored in task offloading for achieving different goals (Peng et al., 2024; Zhang et al., 2024b; Yu et al., 2023; Tan et al., 2024). However, AI algorithms' accuracy for finding optimum edge computing policies relies intensely on solid computing power and an extensive training set that can be difficult to obtain in resource-constrained network edge and dynamic wireless environments. Moreover, the varying task generation properties and the dynamic nature of a wireless network may trigger new training processes frequently; the non-negligible training time would prolong the decision time. In addition, the RIS-assisted task offloading problem involves various types of decisions in various network elements (i.e., offloading decisions in smart devices, reconfigurable phase shift matrices of the RIS, computation resource allocation in MEC servers, etc.), the existing AI-based algorithms, e.g., Tan et al. (2024), Zhang et al. (2024b), require to decompose the problem into at least two subproblems and use iterative methods to obtain optimum solutions. In particular, the previous proposals only use AI algorithms to solve partial decisions. Finally, due to data-driven, the decisions given by present AI-based algorithms are usually uninterpretable and might be unreliable due to the low quality of data samples. Therefore, a low-complex and interpretable solver deserves further study.

The novelties of this paper over existing work include: (1) The active RIS-assisted end-edge collaborative task partitioning and offloading are studied for delay-sensitive and computation-intensive industrial applications, where an active RIS is deployed between IoT users and MEC servers to enhance task partial offloading. (2) We theoretically analyze the delay-guaranteed and energy-efficient properties of the joint decisions, including task partitioning and offloading ratios, active RIS phase shift matrices and amplification factors, and MEC resource allocation, for the problem. (3) A low-complexity and interpretable scheme (termed RISADA) is proposed to solve the problem based on the theoretical analysis.

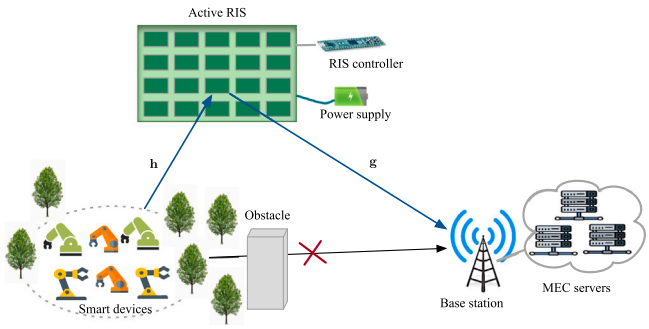

Fig. 1. Active RIS-assisted industrial edge computing system (high-rise buildings and trees block direct links).

3. Model formulation

3.1. System model

This paper focuses on an active RIS-assisted industrial edge computing system where SDs (e.g., machines, industrial terminals, sensor monitors) are located in an industrial factory. In contrast, MEC servers are located in the center of a wireless network. The MEC servers form a virtual pool of computing resources through virtualization technology to provide computing services for offloading tasks. SDs communicate with MEC servers via a base station (BS), as illustrated in Fig. 1. It is assumed that high-rise buildings and trees block the direct links from SDs to the base station. Therefore, an active RIS is placed between SDs and the BS to enhance communication efficiency via reconfiguring the phase shift matrices and amplifying the reflected signal (Zhang et al., 2022; Zhi et al., 2022). Considering their different computation capabilities, an SD only computes one task at a time, while the MEC servers can process multiple tasks in parallel via virtualization technologies.

Assume that time is slotted and the length of a slot is long enough for processing a large task/subtask. Assume that an SD at most generates one industrial computation task in a time slot. The number of SDs generating tasks varies with time slots. Each task could be partitioned into multiple subtasks, each with an independent data segment, which captures the industrial applications requiring objective recognition, such as factory environment monitoring, defective product detection, motion recognition for industrial robots, etc. In this application, multimedia data taken by cameras and other sensors deployed in a factory can be partitioned into multiple data segments and processed in SDs and MEC servers, respectively. Therefore, for each task, the SD has to make a task partition decision, that is, to decide how to partition the task into subtasks and allocate them for edge computing and local computing.

Two other types of decisions need to be made for the subtasks determined to edge computing: (a) task upload-related decisions, which include the transmission power, the RIS's phase shift matrices, and amplification factors, for energy-efficiently reducing the task upload delay; (b) MEC computation resource allocation, which decides how to allocate the MEC resource among offloaded subtasks to reduce the task computation delay.

The number of tasks from SDs to compete for the shared wireless and computation resources in a time slot is represented by M.[1] The corresponding set is defined as $\mathcal{M} = \{1, 2, \ldots, M\}$. The number of reflecting elements in the active RIS is represented by N, and the corresponding set is represented by $\mathcal{N}$. The channel coefficients from

[1] For simplicity, this paper considers the joint decisions within a time slot under the assumption that the length of a slot is long enough for processing a large task/subtask. We will consider time slot-continuous decisions in our future work.

Table 1
Notations.

Symbols	Definition
$\mathcal{M}$	SD set
$\mathcal{N}$	Reflecting element set in the active RIS
M	SD number
N	Reflecting element number
$\mathbf{h}$	Channel coefficient (from SDs to active RIS)
$\mathbf{g}$	Channel coefficient (from active RIS to BS)
p_m	The mth SD's transmission power
B	The bandwidth of the wireless link
Θ	The reflection matrix
ρ	The amplification factor
Φ	The RIS's phase shift
σ_r	The noise power introduced by the active RIS
σ	The noise power of the wireless network
γ_m	SNR
R_m	The mth SD's transmission rate
K_m	The multiples of basic unit
S	The basic unit (in bits) of data
W	The basic unit (in processing cycles) of data for computing
α_m	The ratio of the mth task for offloading
f^{E}	The edge computing capability
f_m^{E}	The computation rate allocated to the mth task
f_m^{M}	The mth SD's computing capability
D_m^{O}	The offloading delay provided to the mth task
$D_m^{\mathrm{O,\ Tx}}$	The upload delay
$D_m^{\mathrm{O,\ CPU}}$	The MEC computation delay
D_m^{L}	Local computation delay
D_m	The latency of the mth task
d_m^{Th}	The delay bound of task m
E_m	The energy consumed for processing task m
E_m^{L}	The energy consumed by local computing
$E_m^{\mathrm{O,\ Tx}}$	The energy consumed by subtask upload
$E_m^{\mathrm{O,\ RIS}}$	The energy consumed at the active RIS
$E_m^{\mathrm{O,\ CPU}}$	The energy consumed by MEC computing
E	The system's energy consumption
ρ^{Th}	The maximum amplification of the active RIS
p^{Th}	The maximum transmit power of an SD

SDs to the active RIS, from the active RIS to the base station, are represented by $\mathbf{h} \in \mathbb{C}^{N\times M}$ and $\mathbf{g} \in C^{1\times N}$, respectively. Let $(\cdot)^H$ denote the conjugate transpose of $(\cdot)$. Since the channel state information (CSI) could be estimated through methods described in Zheng et al. (2022), Wei et al. (2021a,b), this paper assumes that the channel coefficients are perfectly known.

The main notations are listed in Table 1.

3.2. Active RIS communication model

Unlike passive RIS, an amplification device is integrated into an active RIS to amplify the signal, as shown in Fig. 1. An attached intelligent RIS controller controls the phase shift and amplification factor (Shi et al., 2023). This paper tries to improve the computation offloading efficiency by optimizing the phase shift matrix and amplification factors. The optimum decisions are made by the policy controller and then distributed to the active RIS via the RIS controller.

Let $\Theta = \mathrm{diag}\{\rho_1 e^{j\theta_1}, \rho_2 e^{j\theta_2}, \ldots, \rho_N e^{j\theta_N}\}$ be the reflection matrix, where θ_n $(n \in \{1,2,\ldots,N\})$ represent the phase shift of the nth reflecting element, $\rho_n > 1$ is the nth amplification factor. Similar to Zhi et al. (2022), we set $\rho_n = \rho$, $\forall n$ for simplicity. Let $\Phi \triangleq \mathrm{diag}\{e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_N}\}$. Then, we have $\Theta = \rho\mathrm{diag}\{e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_N}\} = \rho\Phi$. Thus, considering the quasi-static Rayleigh fading channel, the received signal y_m of BS from the mth SD could be modeled as

$$y_m = \underbrace{\rho\mathbf{g}\Phi\mathbf{h}_m w_m x_m}_{\text{desired signal}} + \underbrace{\rho\mathbf{g}\Phi\mathbf{v}}_{\text{noise introduced by active RIS}} + \underbrace{n_m}_{\text{AWGN noise}}, \tag{1}$$

where $x_m \sim \mathcal{CN}(0,1)$ is the transmitted symbol from the mth SD with $\mathbb{E}(x_m) = 1$; w_m is the beamforming from the mth SD for symbol x_m; $\rho\Phi\mathbf{v}$ represent the noise introduced by active RIS with $\mathbf{v}$ for $\mathbf{v} \sim \mathcal{CN}(0, \sigma_r^2\mathbf{I}_N)$; $n_m \sim \mathcal{CN}(0,\sigma^2)$ is the additive white Gaussian noise (AWGN) (Zhang et al., 2022).

Therefore, the signal-to-noise-ratio (SNR) is expressed by

$$\begin{aligned}\gamma_m &= \frac{p_m\rho^2|\mathbf{g}\Phi h_m|^2}{\rho^2\sigma_r^2\|\mathbf{g}\Phi\|^2 + \sigma^2} \\ &= \frac{p_m|\mathbf{g}\Phi h_m|^2}{\sigma_r^2\|\mathbf{g}\Phi\|^2 + \sigma^2/\rho^2},\end{aligned} \tag{2}$$

where p_m represents the mth SD's transmission power.

Accordingly, the achievable transmission rate of SD m is expressed by

$$R_m = B\log_2\left(1+\gamma_m\right). \tag{3}$$

In an active RIS-assisted communication model, each active RIS element has additional power consumption for the phase shift switch and control circuit. According to Zhi et al. (2022), the additional power consumed by the active RIS is expressed by

$$Q^{\mathrm{RIS}} = N \times (P_{\mathrm{sw}} + P_{\mathrm{dc}}), \tag{4}$$

where P_{sw} and P_{dc} represent the power consumed by the phase shift switch and control circuit, respectively.

3.3. Task partitioning model

For reducing algorithm complexity, this paper assumes that each task could be represented by $K_m S$ and $K_m W$, respectively, where S (in bits) and W (in processing cycles) are the base units of data for data transmission and computing respectively, e.g., W is the batch size for a round of learning in objective recognition and S is the corresponding data size in bits. $K_m \in \mathbf{R}^+$ is the multiples of the basic unit. Accordingly, the mth ($m \in \mathcal{M}$) task could be partitioned into K_m subtasks at most. To fully exploit the impact of active RIS on end-edge collaborative computation offloading for latency reduction and energy efficiency, the subtasks from the same task could be, at most, grouped into two sets executed in local SD and MEC servers, respectively.

Let α_m $(0 \le \alpha_m \le 1)$ be the offload ratio of the mth task. Then, the ratio of the subtasks processed locally could be expressed by $(1-\alpha_m)$. Accordingly, the task size of the mth task for offloading and local computing could be respectively represented by $\alpha_m K_m W$ and $(1-\alpha_m K_m W)$, the data size requiring upload is expressed by $\alpha_m K_m S$.

3.4. Computing model

3.4.1. MEC computing model

When $\alpha_m > 0$, a set of subtasks from the mth SD will be offloaded to MEC servers via the active RIS-assisted wireless network. In this case, we consider two types of delays: the task upload delay $D_m^{\mathrm{O,\ tx}}$ and the MEC computation delay $D_m^{\mathrm{O,\ CPU}}$. Therefore, the offloading delay D_m^{O} of the subtask set from the mth SD could be expressed by

$$D_m^{\mathrm{O}} = D_m^{\mathrm{O,\ Tx}} + D_m^{\mathrm{O,\ CPU}}, \tag{5}$$

where the upload delay is the time duration for transmitting the subtask set from the data source to MEC servers via the active RIS-assisted wireless network, which could be expressed by

$$D_m^{\mathrm{O,Tx}} = \frac{\alpha_m K_m S}{R_m} = \frac{\alpha_m K_m S}{B\log_2\left(1+\frac{p_m|\mathbf{g}\Phi h_m|^2}{\sigma_r^2\|\mathbf{g}\Phi\|^2+\sigma^2/\rho^2}\right)}. \tag{6}$$

The MEC computation delay could be expressed by

$$D_m^{\mathrm{O,\ CPU}} = \frac{\alpha_m K_m W}{f_m^{\mathrm{E}}}, \tag{7}$$

where f_m^{E} is the allocated central/graphics processing unit (PU) rate, which should satisfy

$$\sum_{m\in\mathcal{M}} f_m^{\mathrm{E}} \le f^{\mathrm{E}}, \tag{8}$$

where f^{E} is the total computation rate of the MEC servers provided to the factory.

3.4.2. Local computing model

When $(1-\alpha_m) > 0$, a set of subtasks will be computed locally in the mth SD. For local computing, this set of subtasks only experiences local computation delay. Accordingly, the delay of the local computing at the mth SD could be expressed by

$$D_m^{\mathrm{L}} = \frac{(1-\alpha_m)K_m W}{f_m^{\mathrm{M}}}, \tag{9}$$

where f_m^{M} is the computation capability of the mth SD.

3.4.3. Latency of the whole task

Generally, the relationship between the partitioned subtasks could be independent and dependent, respectively. In the independent case, the subtasks could be executed in parallel, which captures the scenario of an industrial video clip segmented into multiple episodes and separately executed in local SDs and MEC servers. In the latter case, the subtasks should be executed in sequence. For example, when a split learning-based CNN algorithm is used for product defect detection, the learning model (e.g., CNN) is partitioned into two or more parts, and each part consists of several consecutive CNN layers. Different parts will be offloaded to different computing nodes. Then, the associated computing nodes train the model parts in a sequential order (Lin et al., 2024). Since this paper focuses on fully exploiting the communicational and computational capabilities of an active RIS-assisted end-edge collaborative MEC system, we consider the independent case for the partitioned subtasks. In future work, we will extend to the dependent case.

Therefore, the latency of the whole task should be the latency of the latest completed subtask. Accordingly, for the mth task, the latency is expressed by

$$D_m = \max(D_m^{\mathrm{L}}, D_m^{\mathrm{O}}). \tag{10}$$

3.5. Energy consumption model

This paper considers four types of energy consumption, including the energy consumed in SDs for local task computing E_m^{L}, energy consumed in SDs for transmitting the offloaded task $E_m^{\mathrm{O,\ Tx}}$, energy consumed by phase-shift switches and control circuits on active RIS $E_m^{\mathrm{O,\ RIS}}$, and the energy consumed in MEC servers for edge computing $E_m^{\mathrm{O,\ CPU}}$. Notice that the latter three types of energy are consumed for edge computing. The total consumed energy for processing the mth task is expressed by

$$E_m = E_m^{\mathrm{L}} + E_m^{\mathrm{O,\ Tx}} + E_m^{\mathrm{O,\ RIS}} + E_m^{\mathrm{O,\ CPU}}. \tag{11}$$

The energy consumed by local computing is mainly determined by the task size and local computing capabilities; that is, it can be expressed by

$$E_m^{\mathrm{L}} = \beta(1-\alpha_m)K_m W f_m^{\mathrm{M}} = \beta D_m^{\mathrm{L}}(f_m^{\mathrm{M}})^2, \tag{12}$$

where β is the energy factor for local computing.

The transmit power and duration determine the energy that the local SD consumes for transmitting the offloaded task. Accordingly, it is expressed by

$$E_m^{\mathrm{O,\ Tx}} = p_m D_m^{\mathrm{O,\ Tx}}. \tag{13}$$

Based on the discussions in Section 3.2, the energy consumed on an active RIS is expressed by

$$E_m^{\mathrm{O,\ RIS}} = Q^{\mathrm{RIS}} D_m^{\mathrm{O,\ Tx}}. \tag{14}$$

Similar to local computing, the energy consumed for edge computing is expressed by

$$E_m^{\mathrm{O,\ CPU}} = \upsilon\alpha_m K_m W f_m^{\mathrm{E}} = \upsilon D_m^{\mathrm{O,CPU}}(f_m^{\mathrm{E}})^2, \tag{15}$$

where υ is the energy factor of edge computing.

Substituting (12)–(15) into (11) and with some calculus, we have

$$E_m = K_m W\left(\beta f_m^{\mathrm{M}} + \alpha_m(\omega_m - \beta f_m^{\mathrm{M}})\right), \tag{16}$$

where $\omega_m \triangleq (p_m + Q^{\mathrm{RIS}})\chi/R_m + \upsilon f_m^{\mathrm{E}}$ and $\chi \triangleq S/W$.

3.6. Utility model for delay guarantee and energy efficiency

Since delay performance and energy consumption are two focus areas in edge computing, this subsection introduces a system utility model to comprehensively evaluate the delay guarantee and energy consumption of task partitioning and offloading in industrial edge computing.

From an intelligent factory's point of view, delay guarantee is a focus of attention. To evaluate the user satisfaction with delay performance, we define the delay-based service satisfaction of SD m for $m \in \mathcal{M}$ as follows.

$$\zeta_m = \mathbf{1}(D_m \le D_m^{\mathrm{Th}}), \tag{17}$$

where D_m^{Th} is the delay bound of task m, $\mathbf{1}(K) = 1$ if k is true; $\mathbf{1}(k) = 0$ otherwise. That is, if the latency D_m of the task does not exceed the delay bound, the SD is satisfied with the task partitioning and offloading service; otherwise, the SD is not satisfied with the service.

From an edge computing service provider's point of view, energy efficiency is a focus of attention.

Since the larger ζ_m is, the more satisfactory the delay guarantee is, and the smaller E_m is, the more satisfactory the energy efficiency is, we define the utility of processing task m as follows.

$$G_m = \zeta_m - \varphi E_m, \tag{18}$$

where φ is a weighting factor to help uniformly evaluate delay performance and energy consumption.

Accordingly, the system utility for processing all tasks is defined by

$$G = \sum_{m\in\mathcal{M}} G_m. \tag{19}$$

4. Problem formulation

4.1. JCORO

This paper aims to maximize system utility through jointly optimizing (a) task partitioning and offloading decisions $\boldsymbol{\alpha} \triangleq \{\alpha_m : \forall m \in \mathcal{M}\}$, (b) phase shift and amplification factor ($\boldsymbol{\Phi}$, ρ) of the active RIS, (c) transmission power $\mathbf{p} \triangleq \{p_m : \forall m \in \mathcal{M}\}$, and (d) computation resource allocation $\mathbf{f} \triangleq \{f_m^{\mathrm{E}} : \forall m \in \mathcal{M}\}$ for offloaded subtasks. The above problem is called the joint computation offloading and active RIS optimization (JCORO) problem, and is formulated as,

$$\text{JCORO}: \max_{\{\alpha,\boldsymbol{\Phi},\rho,\mathbf{p},\mathbf{f}\}} \quad G \tag{20a}$$

$$\text{s.t.} \quad 0 \le \alpha_m \le 1, \forall m \in \mathcal{M}, \tag{20b}$$

$$0 \le \theta_n \le 2\pi, \forall n \in \mathcal{N}, \tag{20c}$$

$$1 < \rho < \rho^{\mathrm{Th}}, \tag{20d}$$

$$p_m \le p^{\mathrm{Th}}, \tag{20e}$$

$$\sum_{m\in\mathcal{M}} f_m^{\mathrm{E}} \le f^{\mathrm{E}}, \tag{20f}$$

where (20) follows (19); (20b) is the partitioning and offloading constraint; (20b) is the phase shift constraint; (20d) is the amplification factor constraint in active RIS; (20e) is the transmission power constraint; (20f) is the computing resource constraint in edge computing.

4.2. Problem decomposition

The JCORO problem described in (20) involves end-user and edge computing network decision-making, e.g., task partitioning and offloading decisions for multiple SDs, phase shift matrix and amplification factors in an active RIS, and computation resource allocation in MEC servers. To reduce algorithm complexity, this paper decomposes JCORO into two concatenated subproblems: the PO and RORA subproblems. The task partitioning and offloading decisions α^* of PO are based on the latest results of RORA. The RORA solution will be affected by the task partitioning and offloading decisions from the PO. The details are described below.

Firstly, given the up-to-date $\{\boldsymbol{\Phi}, \rho, \mathbf{p}\}$ of RORA, the current value of transmission rate R_m ($\forall m \in \mathcal{M}$) could be estimated. Thus, the upload delay is determined. Similarly, given $\mathbf{f}$ of the RORA subproblem, the computation delay could be estimated. Then, the latency of a task D_m ($\forall m \in \mathcal{M}$) is determined. Therefore, from an SD's point of view, JCORO is reduced to PO as follows.

$$(\text{PO}): \max_{\alpha*} \quad G \tag{21a}$$

$$\text{s.t.} \quad (20b), \tag{21b}$$

$$\boldsymbol{\Phi} = \boldsymbol{\Phi}^*, \rho = \rho^*, \mathbf{p} = \mathbf{p}^*, \mathbf{f} = \mathbf{f}^*, \tag{21c}$$

where (20c)–(20f) of JCORO are released given $\{\boldsymbol{\Phi}^*, \rho^*, \mathbf{p}^*, \mathbf{f}^*\}$; (21c) is the up-to-date solver of RORA.

Secondly, RORA's objective is to improve SDs' delay-based service satisfaction by jointly optimizing active RIS and MEC computation resource allocation, considering the latest results of PO. That is, the RORA subproblem is formulated as,

$$(\text{RORA}): \max_{\{\boldsymbol{\Phi}^*, \rho^*, \mathbf{p}^*, \mathbf{f}^*\}} \quad \zeta_m, \forall m \in \mathcal{M} \tag{22a}$$

$$\text{s.t.} \quad (20c),(20d),(20e),(20f), \tag{22b}$$

$$\alpha = \alpha^*, \tag{22c}$$

where (22c) is the solution of the PO subproblem.

5. Theoretical analysis

This section analyzes the properties of decision variables in PO and RORA subproblems, respectively, concerning delay guarantee and energy efficiency.

5.1. Task partitioning and offloading properties

According to (5), (6) and (7), for the mth task, the offloading delay could be expressed by

$$\begin{aligned} D_m^{\mathrm{O}} &= \alpha_m K_m S / R_m + \alpha_m K_m W / f_m^{\mathrm{E}} \\ &= \alpha_m K_m W \psi_m, \end{aligned} \tag{23}$$

where $\psi_m \triangleq \chi / R_m + 1/f_m^{\mathrm{E}}$.

According to (10), the latency of the whole task is derived by

$$\begin{aligned} D_m &= \max(D_m^{\mathrm{L}}, D_m^{\mathrm{O}}) \\ &= \max\Big((1-\alpha_m) K_m W / f_m^{\mathrm{M}}, \alpha_m K_m W \psi_m\Big) \\ &= \max\Big((1-\alpha_m) / f_m^{\mathrm{M}}, \alpha_m \psi_m\Big) \cdot K_m W. \end{aligned} \tag{24}$$

Theorem 1. *For any task $m \in \mathcal{M}$, and under any delay-guaranteed partitioning and offloading policy, the offloading ratio of the task should satisfy*

$$\begin{cases} \alpha_m \geq 1 - \dfrac{D_m^{Th} f_m^M}{K_m W}, \\ \alpha_m \leq \dfrac{D_m^{Th}}{\psi_m K_m W}. \end{cases} \tag{25}$$

The statement of Theorem 1 indicates that the offloading ratio of a task is affected by an SD's computation capability f_m^{M}, the task's tolerable latency D_m^{Th} and computing amount $K_m W$, and the edge processing service provided to this task ψ_m, which is affected by the transmission rate and the allocated MEC computation resource (see (23)). The lower bound of α_m in (25) indicates that the task amount that exceeds the local SD's delay-based computation capability (e.g., $D_m^{\mathrm{Th}} f_m^{\mathrm{M}} / (K_m W)$) must be offloaded. In contrast, the upper bound of α_m in (25) indicates that the offloading ratio could not exceed the allocated edge processing service.

The statement of Theorem 1 also indicates that if the task amount and local computation capability are given, there are tradeoffs between the task's tolerable latency, transmission rate, and MEC computation resource allocation. The larger the tolerable latency, the smaller the transmission rate and MEC computation resource requirements, and vice versa. Once the delay bound is given, the smaller the transmission rate, the larger the MEC computation resource should be allocated for delay guarantee.

Lemma 1. *For any task $m \in \mathcal{M}$, and under any delay-guaranteed partitioning and offloading policy, if the latest completed subtask is from local computing, that is, If $D_m = D_m^L$, then the dominated task offloading ratio α_m^* satisfies*

$$\begin{cases} \alpha_m^* \geq \alpha_m^{lb\_l}, & (26) \\ \alpha_m^* \leq \alpha_m^{ub\_l}, & (27) \end{cases}$$

where $\alpha_m^{lb\_l} \triangleq 1 - \frac{D_m^{Th} f_m^M}{K_m W}$, and $\alpha_m^{ub\_l} \triangleq \min\Big(\frac{D_m^{Th}}{\psi_m K_m W}, \frac{1}{1+\psi_m f_m^M}\Big)$.

The optimum system utility for processing the corresponding task satisfies

$$G_m^* \leq \begin{cases} \varphi - \varphi_2 K_m W \left(\beta f_m^M + \alpha_m^{lb\_l}(\omega_m - \beta f_m^M)\right), \omega_m \geq \beta f_m^M \\ \varphi - \varphi_2 K_m W \left(\beta f_m^M + \alpha_m^{ub\_l}(\omega_m - \beta f_m^M)\right), otherwise. \end{cases} \tag{28}$$

where $\varphi_2 = \frac{(1-\varphi)}{E^{max}}$.

Lemma 2. *For any task $m \in \mathcal{M}$, and under any delay-guaranteed partitioning and offloading policy, if the latest completed subtask is from MEC computing, that is, If $D_m = D_m^O$ holds, then the dominated task offloading ratio α_m^* satisfies*

$$\begin{cases} \alpha_m^* \geq \alpha_m^{lb\_o}, & (29) \\ \alpha_m^* \leq \alpha_m^{ub\_o}, & (30) \end{cases}$$

where $\alpha_m^{lb\_o} \triangleq \max\Big(1 - \frac{D_m^{Th} f_m^M}{K_m W}, \frac{1}{1+\psi_m f_m^M}\Big)$, and $\alpha_m^{ub\_o} \triangleq \frac{D_m^{Th}}{\psi_m K_m W}$.

The optimum utility for processing the corresponding task satisfies

$$G_m^* \leq \begin{cases} \varphi - \varphi_2 K_m W \left(\beta f_m^M + \alpha_m^{lb\_o}(\omega_m - \beta f_m^M)\right), \omega_m \geq \beta f_m^M \\ \varphi - \varphi_2 K_m W \left(\beta f_m^M + \alpha_m^{ub\_o}(\omega_m - \beta f_m^M)\right), otherwise. \end{cases} \tag{31}$$

The statements of Theorem 1 and its extensions (e.g., Lemmas 1–2) are generalized to dynamic and heterogeneous network conditions; that is, the offloading ratio varies with varying computation amount, delay bound, transmission rate, and allocated computation resource.

Theorem 2. *For any task from SD $m \in \mathcal{M}$, and under any task partitioning and offloading policy, if $D_m \geq D_m^{Th}$, then the dominated partitioning and offloading decision satisfies*

$$\alpha_m^* = \begin{cases} 0, \omega_m \geq \beta f_m^M & (32) \\ 1, otherwise. & (33) \end{cases}$$

The optimum utility for processing the task satisfies

$$G_m^* = -(1-\varphi) E_m^* / E^{max}, \tag{34}$$

where $E_m^* = K_m W \left(\beta f_m^M + \alpha_m^*(\omega_m - \beta f_m^M)\right)$.

When system load is heavy, e.g., the total computation demand is close to or exceeds the system computation capacity (which is the sum of SDs' and MEC servers' computation resources), or the available bandwidth cannot satisfy the transmission rate requirement of offloading tasks, the latency of some task may exceeds its delay bound, i.e., $D_m \geq D_m^{\mathrm{Th}}$ for $m \in \mathcal{M}$. In this case, from a utility optimization point of view, the statement of Theorem 2 indicates that it is better to process the whole task locally or in an edge server, depending on which action minimizes energy consumption.

5.2. Joint active RIS optimum and MEC computation resource allocation

Theorem 3. *For any task $m \in \mathcal{M}$, given α_m, then, under any delay-guaranteed RIS optimum and MEC resource allocation policy, the transmission rate of the offloading subtasks should satisfy*

$$R_m \geq \frac{\chi}{\frac{D_m^{Th}}{K_m W \alpha_m} - \frac{1}{f_m^E}}, \tag{35}$$

and the transmission power should satisfy

$$\begin{cases} p_m \geq \left(ln\left(\frac{\chi}{B(\frac{D_m^{Th}}{K_m W \alpha_m} - \frac{1}{f_m^E})} \right) + 1 \right) \frac{\sigma_r^2 \|\mathbf{g}\boldsymbol{\Phi}\|^2 + \sigma^2/\rho^2}{|\mathbf{g}\boldsymbol{\Phi} h_m|^2}, & (36) \\ p_m \leq p^{Th}. & (37) \end{cases}$$

Theorem 4. *For any task $m \in \mathcal{M}$, given α_m and R_m, then, under any delay-guaranteed RIS optimum and MEC resource allocation policy, the allocated MEC computation resource should satisfy*

$$\begin{cases} f_m^E \geq \frac{1}{\frac{D_m^{Th}}{K_m W \alpha_m} - \frac{\chi}{R_m}}, & (38) \\ \sum_{m \in \mathcal{M}} f_m^E \leq f^E. & (39) \end{cases}$$

Lemma 3. *For any task $m \in \mathcal{M}$, given α_m and R_m, then, under any delay-guaranteed RIS optimum and MEC resource allocation policy, if $\omega_m \geq \beta f_m^M$ holds, then the optimum allocated MEC computation resource f_m^{E*} satisfies*

$$\begin{cases} f_m^{E*} \geq \max(f_m^{lb\_a}, f_m^{lb\_b}), & (40) \\ f_m^{E*} + \sum_{m' \in \mathcal{M} \setminus \{m\}} f_{m'}^E \leq f^E. & (41) \end{cases}$$

where $f_m^{lb\_a} = \frac{1}{\frac{D_m^{Th}}{K_m W \alpha_m} - \frac{\chi}{R_m}}$, $f_m^{lb\_b} = \frac{1}{v}(\beta f_m^M - (p_m + Q^{RIS})\chi / R_m)$.

Otherwise, the optimum allocated MEC computation resource f_m^{E*} satisfies

$$\begin{cases} f_m^{E*} \geq \frac{1}{\frac{D_m^{Th}}{K_m W \alpha_m} - \frac{\chi}{R_m}}, & (42) \\ f_m^{E*} \leq \min(f_m^{lb\_b}, f_m^{ub}). & (43) \end{cases}$$

where $f_m^{ub} = f^E - \sum_{m' \in \mathcal{M} \setminus \{m\}} f_{m'}^E$, *we get the optimum* G_m^*.

The proof of the above theorems and lemmas can be found in Appendix.

6. RISADA scheme

Based on the analytical results, this section designs an active RIS-assisted delay-aware (RISADA) task partitioning and offloading scheme to address the JCORO problem described in (20). The details are as follows.

6.1. Overview of RISADA

The analytical results in Lemmas 1–2 show that, for each task, the source of the latest completed subtask, e.g., from local computing or MEC computing, affects the task partition and offload decision. Since the latest completed subtask is affected by the active RIS (affecting the transmission rate) and MEC resource allocation policies (affecting the MEC computation delay), once the active RIS and MEC resource allocation decisions are given, the optimum task partitioning and offloading decisions could be obtained via Lemmas 1–2. On the other hand, once the task partitioning and offloading decisions are given, the required transmission rate, transmission power, and MEC computation resource could be determined via Theorem 3 and Lemma 3. Accordingly, we design a two-stage iterative RISADA scheme to solve the JCORO problem.

As shown in Algorithm 1, we initially set the task partitioning and offloading decision α as a random vector. We find out the optimum strategy set $\mathcal{A}^* = (\alpha^*, \boldsymbol{\Phi}^*, \rho^*, \mathbf{p}^*, \mathbf{f}^*)$ with the following steps: (1) In stage-A, we solve the RORA subproblem. In particular, we use a `stochastic initialized` and gradually approximated sum-SNR-maximization beamforming optimization (SIG) to find out optimum amplification factor and phase shift $(\rho, \boldsymbol{\Phi})$ based on the present offloading decision α via Algorithm 2; Then, we adjust the transmission power and estimate the transmission rate (p_m, R_m) adaptive to present offloading decision and RIS configuration; Next, we allocate the MEC computation resource $\mathbf{f}$ to offloaded subtasks based on the present offloading decision and transmission rate; (2) In Stage-B, we solve PO and obtain the updated task partitioning and offloading decisions α as well as estimate system utility G with present transmission and computation resource allocation decisions; (3) If the estimated system utility G is improved, then the task partitioning and offloading strategy will be updated. We repeat steps (1) to -(3) until no update happens or the repeat count reaches the threshold. After obtaining the optimum strategy, the tasks are processed based on the strategy.

6.2. SIG for beamforming optimization

The position of the RIS affects the communication efficiency between the SDs and the BS. Generally, the smaller the distance of the user-RIS or RIS-BS, the greater the communication improvement. Accordingly, this paper assumes that the RIS is deployed within the industrial factory close to SDs. However, due to distinct incident angles and positions of SDs, different SDs may yield different communication efficiencies in terms of transmission rate in a multi-user multi-reflecting element system.

Since the distances of user-RIS and RIS-BS are constant in a decision epoch, the transmission rate is affected by the phase shift, amplification factor of the active RIS, and the transmit power of an SD, as illustrated in (2) and (3). Each SD could improve its transmission rate by increasing the active RIS's transmission power and amplification factor at the cost of energy consumption. However, each reflecting element in RIS could be reconfigured with one phase shift value at the moment, leading to distinct signal reflecting strength provided to different SDs due to their distinct incident angles. Therefore, releasing transmission power and amplification factor, the beamforming optimization could be transferred to a sum SNR maximization problem by re-configuring the phase shift (called PhaseOpt), which is formulated as,

$$\begin{aligned} \text{PhaseOpt}: \max_{\boldsymbol{\Phi}^*} \quad & \gamma = \sum_{m \in \mathcal{M}^{\mathrm{O}}} \gamma_m & (44a) \\ \text{s.t.} \quad & \gamma_m = |\mathbf{g}\boldsymbol{\Phi} h_m|^2, & (44b) \\ & 0 \leq \theta_n \leq 2\pi, \forall n \in \mathcal{N} & (44c) \end{aligned}$$

where (44b) follows (2), (44c) follows ((20c)).

For any $m \in \mathcal{M}^{\mathrm{O}}$, to maximize γ_m is equivalent to maximize $|\mathbf{g}\boldsymbol{\Phi} h_m|^2$. Let $\mathrm{v} = (e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_N})^H$, then we have $\mathbf{g}\boldsymbol{\Phi} h_m = \mathrm{v}^H \mathrm{diag}(\mathbf{g}) h_m$.

Algorithm 1 RISADA

Initialization: $\alpha_m = \epsilon$, $G^{\text{opt}} = 0, f_m^{\text{E}} = f^{\text{E}}/M$, $f_m^{\text{M}} = f^{\text{M}}$ for $m \in \mathcal{M}$, where ϵ is a random number range from 0 to 1.

Finding out optimum policy:

1. **Stage-A:** Solve the RORA subproblem using the following steps:
   a. Determine $(\rho, \boldsymbol{\Phi})$ using Algorithm 2.
   b. Determine $(\mathbf{p}, \mathcal{R})$ for offloading subtasks based on Theorem 3.
   c. Allocate computation resource $f_m^{\text{E}}, \forall m \in \mathcal{M}^{\text{O}}$ to offloading subtasks based on Lemma 3, set $f_m^{\text{E}} = 0$ for $m \in \mathcal{M} \setminus \mathcal{M}^{\text{O}}$, set $\mathbf{f} = \{f_m^{\text{E}} : \forall m \in \mathcal{M}\}$.
2. **Stage-B:** Solve the PO subproblem to obtain $\alpha = \{\alpha_m : \forall m \in \mathcal{M}\}$ using Algorithm 3 and calculate system utility G with present $(\mathcal{R}, \mathbf{f})$ via (19).
3. **Policy update:** If $G > G^{\text{opt}}$, then set the latest $\mathcal{A} = (\alpha, \boldsymbol{\Phi}, \rho, \mathbf{p}, \mathbf{f})$ as $\mathcal{A}^* = (\alpha^*, \boldsymbol{\Phi}^*, \rho^*, \mathbf{p}^*, \mathbf{f}^*)$, set $G^{\text{opt}} = G$.
4. **Repeat** steps 1)-3) until no updated $\mathcal{A} = (\alpha, \boldsymbol{\Phi}, \rho, \mathbf{p}, \mathbf{f})$ could improve G or the repeat count reaches the threshold.
5. **output:** $\mathcal{A}^* = (\alpha^*, \boldsymbol{\Phi}^*, \rho^*, \mathbf{p}^*, \mathbf{f}^*)$.

Processing:

1. Set the phase shift and amplification factor of the active RIS with $(\boldsymbol{\Phi}^*, \rho^*)$.
2. Partition tasks with α^*;
3. Perform local computing based on α^* and $f_m^{\text{M}^*}$ for $m \in \mathcal{M}$.
4. Offload the partitioned tasks to the network edge for MEC computing with $\mathbf{p}^*$ and $\mathbf{f}^*$.

Accordingly, the problem described in (44) could be transformed into

$$\max_{\mathbf{v}} \quad \sum_{m \in \mathcal{M}^{\text{O}}} |\mathbf{v}^H \text{diag}(\mathbf{g}) h_m|^2 \tag{45a}$$

$$\text{s.t.} \quad |v_n| = 1, \forall n = 1, 2, \ldots, N, \tag{45b}$$

$$\arg(\mathbf{v}^H \text{diag}(\mathbf{g}) h_m) = 0, \forall m \in \mathcal{M}^{\text{O}}, \tag{45c}$$

where the right-hand side of (45c) indicates that when the equivalent link of a reflecting path equals a direct link, we get the strongest reflecting signal.

For $m \in \mathcal{M}^{\text{O}}$, the optimum $|\mathbf{v}^H \text{diag}(\mathbf{g}) h_m|^2$ is yielded by $\mathbf{v}_m^* = e^{j(-\arg(\text{diag}(\mathbf{g}) h_m))}$. Thus, considering the user-specific channel quality, the nth phase shift is given by

$$\theta_{m,n} = -\arg(g_n h_{n,m}) = -\arg(g_n) - \arg(h_{n,m}), \tag{46}$$

where g_n is the nth element of $\mathbf{g}$, $h_{n,m}$ is the channel coefficient.

Based on the above discussions, we propose SIG to determine the optimum ρ and $\boldsymbol{\Phi}$. The detail is shown in Algorithm 2.

6.3. Delay-aware power allocation

To guarantee the delay requirement, the transmission rate of the offloading subtasks should be no less than the lower bound, as illustrated in Theorem 3. In addition, to satisfy the transmission rate requirement, the required transmission power should satisfy (36). Accordingly, in Stage-A.b of Algorithm 1, given the up-to-date $(\rho, \boldsymbol{\Phi}, \alpha_m, f_m^{\text{E}})$, the transmission power of the mth $(\forall m \in \mathcal{M})$ subtask is set to

$$p_m = \min\left(p_m^{\min} + \epsilon, p^{\text{Th}}\right), \tag{50}$$

Algorithm 2 SIG for beamforming optimization

Input: N, h^{O}, $\mathbf{g}$, $\mathcal{M}^{\text{O}}$, ρ^{Th}.
Output: ρ^*, $\boldsymbol{\Phi}^*$.
Initiate: set $\rho^* = \rho^{\text{Th}}$.
For $L = 1$ to $L_{\max}$, do

1. Set $\theta_n = \epsilon_\pi$ for $n \in \mathcal{N}$, where ϵ_π is a random value range from 0 to 2π, set $w_m = \frac{(\mathbf{g}\boldsymbol{\Phi} h_m)^H}{\|h_m\|}$ for $m \in \mathcal{M}^{\text{O}}$, where $\mathbf{g}$ is a $1 \times N$ vector, $\boldsymbol{\Phi} = \text{diag}(e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_N})$, h_m is a $N \times 1$ vector.
2. **For** $n \in \mathcal{N}$, do
   (a) Calculate $\mathbf{h}_n$ with $\mathbf{h}_n = h_n w$, where h_n is a $1 \times M^{\text{O}}$ vector, $w = (w_m : \forall m \in \mathcal{M}^{\text{O}})$ is a $M^{\text{O}} \times 1$ vector.
   (b) Calculate θ_{h} with $\theta_{\text{h}} = \arg(\mathbf{h}_n)$, where $\arg(\cdot)$ represents the phase of $(\cdot)$.
   (c) Calculate θ_{g} with $\theta_{\text{g}} = \arg(g_n)$.
   (d) Calculate the nth phase shift using (47).

$$\theta_n' = -\theta_{\text{h}} - \theta_{\text{g}}. \tag{47}$$

   (e) Calculate $\boldsymbol{\Phi}'$ using (48).

$$\boldsymbol{\Phi}' = \text{diag}(e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_n'}, \ldots, e^{j\theta_N}). \tag{48}$$

   In (48), the phase shift $\theta_{n'}$ of the n'th (for $n' \in \mathcal{N} \setminus \{n\}$) element is the phase shift in the previous round.
   (f) Update θ_n, $\boldsymbol{\Phi}$ and H_m for all $m \in \mathcal{M}^{\text{O}}$ as follows.

$$\begin{cases} \theta_n = \theta_n', \\ \boldsymbol{\Phi} = \boldsymbol{\Phi}', \\ w_m = \frac{(\mathbf{g}\boldsymbol{\Phi} h_m)^H}{\|\mathbf{g}\boldsymbol{\Phi} h_m\|}. \end{cases} \tag{49}$$

3. Calculate $\gamma' \triangleq \sum_{m \in \mathcal{M}^{\text{O}}} |\mathbf{g}\boldsymbol{\Phi} h_m|^2$ according to (2).
4. If $\gamma' > \gamma$, then set $\boldsymbol{\Phi}^* = \boldsymbol{\Phi}$ and $\gamma = \gamma'$.

where $L_{\max} \le M^{\text{O}}$ is a repeat threshold.

where $p_m^{\min} = \left(\ln\left(\frac{\chi}{B(\frac{D_m^{\text{Th}}}{K_m W \alpha_m} - \frac{1}{f_m^{\text{E}}})}\right) - 1\right) \frac{\sigma_r^2 \|\mathbf{g}\boldsymbol{\Phi}\|^2 + \sigma^2/\rho^2}{|\mathbf{g}\boldsymbol{\Phi} h_m|^2}$ according to (36), $\epsilon > 0$ is a random number. Once ρ, $\boldsymbol{\Phi}$ and p_m are determined, the optimum transmission rate R_m could be yielded via (3). Set $\mathbf{p} = \{p_m : \forall m \in \mathcal{M}\}$ and $\mathcal{R} = \{R_m : \forall m \in \mathcal{M}\}$, then we obtain optimum $(\mathbf{p}, \mathcal{R})$ for offloading subtasks.

6.4. DEEPO for task partitioning and offloading

The analytical results in Theorem 1 show that, given the transmission rate and MEC computation resource (e.g., ψ_m, f_m for $m \in \mathcal{M}$), for guaranteeing the delay, the offloading ratio of the task should satisfy (25). Moreover, for achieving optimum system utility, that is, for energy-efficiently guaranteeing the delay, the offloading ratio is further constrained by the latest completed subtask, as illustrated in Lemmas 1–2. In addition, if the network and computation resource cannot satisfy a task's delay requirement, then the offloading ratio is different from Lemmas 1–2, as illustrated in Theorem 2. Based on the above analysis, we design the Delay-aware energy-efficient task partitioning and offloading (DEEPO) algorithm in stage-B to determine the optimum policy.

The detail is shown in Algorithm 3.

6.5. Algorithm complexity

This paper decomposes the JCORO problem into two concatenated subproblems: the PO and RORA subproblems. The proposed RISADA

Algorithm 3 DEEPO for the PO subproblem

Input: R_m, f_m^{E}, f_m^{M}, α_m for $m \in \mathcal{M}$.
Output: α_m^* for $m \in \mathcal{M}$.
Initiate: Calculate G with input $(\alpha, \mathcal{R}, \mathbf{f})$.
Repeat

1. Find out a set of tasks, named $\mathcal{M}'$, whose task partitioning and offloading ratio is overflow, that is, the present α_m for $m \in \mathcal{M}'$ does not satisfy Lemmas 1–2.
2. Find out the task m^* whose task partitioning and offloading decision update could significantly improve the system utility from the overflow set $\mathcal{M}'$, which further includes the following steps.

   - **Initial:** Let $\Delta G = 0$, $m^* = \{\}$, $\alpha_{m*} = 0$.
   - **For** $m \in \mathcal{M}'$, do

     (a) Calculate D_m^{L} and D_m^{O} with (9) and (5), respectively.
     (b) Update α'_m. In special, if $D_m^{\mathrm{L}} > D_m^{\mathrm{O}}$, then, according to Lemma 1, update α'_m with (51).

$$\alpha'_m = \alpha_m^{\mathrm{lb\_l}} + (\alpha_m^{\mathrm{ub\_l}} - \alpha_m^{\mathrm{lb\_l}})\epsilon, \tag{51}$$

     else, according to Lemma 2, update α'_m with (52).

$$\alpha'_m = \alpha_m^{\mathrm{lb\_o}} + (\alpha_m^{\mathrm{ub\_o}} - \alpha_m^{\mathrm{lb\_o}})\epsilon, \tag{52}$$

     where ϵ is a random number between 0 and 1.
     (c) Let $\alpha' = (\alpha_1, \ldots, \alpha_{m-1}, \alpha'_m, \alpha_{m+1}, \ldots)$, calculate G' based on α' with (18).
     (d) Calculate $\Delta G'$ with $\Delta G' = G' - G$.
     (e) If $\Delta G' > \Delta G$, then update G, ΔG, m^* and α_{m*} as follows

$$\begin{cases} G = G', \\ \Delta G = \Delta G', \\ m^* = m, \\ \alpha_{m*} = \alpha'_m. \end{cases} \tag{53}$$

Until no update can improve G or reaches the repeat count.

scheme solves these two subproblems with two iterative stages. In stage-A, we solve the RORA subproblem by (1) using SIG (Algorithm 2) to find out the optimum amplification factor and phase shift $(\rho, \boldsymbol{\Phi})$, where the algorithm complexity of SIG is $O(M^{\mathrm{O}} \times N)$; (2) determining $(\mathbf{p}, \mathcal{R})$ based on Theorem 3, the algorithm complexity is $O(1)$; (3) allocating computation resource to offloading subtasks, the corresponding algorithm complexity is $O(\mathcal{M}^{\mathrm{O}})$. Since SIG runs in an RIS controller, the determining of transmission rate runs in a wireless network, and computation resource allocation performs in MEC servers, the algorithm complexity in stage-A for solving RORA is $\max\left(O(M^{\mathrm{O}} \times N), O(1), O(\mathcal{M}^{\mathrm{O}})\right)$, which equals to $O(M^{\mathrm{O}} \times N)$. In stage-B, as shown in Algorithm 3, the algorithm complexity for solving the PO subproblem is $O(M)$. Therefore, the algorithm complexity of RISADA is $O(M^{\mathrm{O}} \times N \times M)$.

It is noted that optimizing the amplification factor and phase shift $(\rho, \boldsymbol{\Phi})$ of an RIS with multiple users is complex (Liu et al., 2024; Lv et al., 2024; Wu and Zhang, 2019). For example, using semidefinite relaxation (SDR) (Wu and Zhang, 2019), the algorithm complexity is $O(N^2)$ for a single-user system. For a multi-user system, under a two-stage alternating optimization-based algorithm (Wu and Zhang, 2019), the algorithm complexity is $O(M \times N)$. With the gradient descent method (Liu et al., 2024), the algorithm complexity is $O(M \times N)$. The above analysis shows that the algorithm complexity for finding out an optimizing phase shift of an RIS is at least a function of the user and reflecting element numbers. The joint optimization of the amplification factor and phase shift of an RIS, task partitioning and offloading, and the computation resource allocation is more complex. In addition, the algorithm complexity of a resource allocation policy with M users/tasks is generally a function of M. Since industrial computation tasks are typically generated on timescales of hours or days and processor power continues to increase, our algorithm is feasible for real-time or near-real-time applications. Accordingly, our algorithm complexity is acceptable, and the proposed algorithm is scalable.

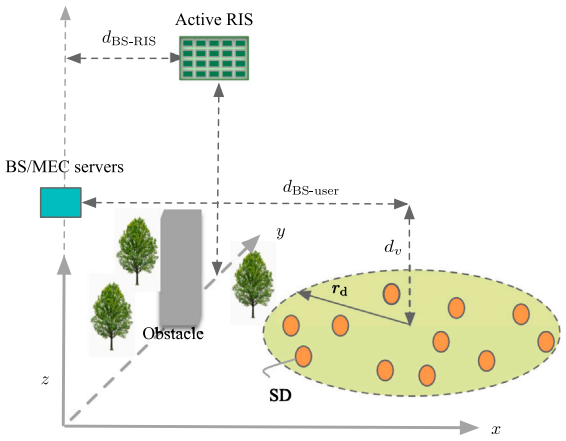

Fig. 2. Simulation setup (top view).

7. Performance evaluation

This section evaluates the effectiveness of our proposal for active RIS-assisted task partitioning and offloading in industrial-edge computing.

7.1. Parameter setting

Consider the production scale and site size of a medium-sized factory, the number of SDs and active RIS reflecting elements are set to $M = 20$ and $N = 400$, respectively. As illustrated in Fig. 2, the positions of the base station and RIS are set to (0,0,0)m and (50,0,10)m, respectively (Wu and Zhang, 2019). SDs are distributed in a circular area with minimum and maximum horizontal (x-axis) distances from the base station are $d_{\min} = 60$ m and $d_{\max} = 80$ m, respectively. The radius of the circular area is set to $r_{\mathrm{d}} = (d_{\max} - d_{\min})/2$. The vertical (z-axis) distance from the base station is set to $d_{\mathrm{v}} = -3$ m. It is assumed that there are obstacles between the SD area and the base station so that direct communication between SDs and the base station is blocked.

The computing resources of the MEC servers and SDs are set to $f^{\mathrm{E}} = 50$ GHz and $f_m^{\mathrm{M}} = 1.2$ GHz for $m \in \mathcal{M}$, respectively. Each SD generates one task per epoch; task sizes are randomly distributed between 1 to 20 times the basic unit, where the basic unit of task size and computing amount for task partition are set to $S = 0.4$ Mb and $W = 0.84$ GHz, respectively, which captures the scenario of industrial image recognition, where the data generated per SD is about 0.7Mb/task (Navarro-Ortiz et al., 2020). The basic parameter settings are listed in Table 2.

7.2. Evaluation of active RIS for transmission rate improvement

This subsection evaluates the efficiency of active RIS for task upload from local SDs to MEC servers for MEC computing. We compare the performance of our proposed beamforming solver for active RIS (e.g., SIG) with sum-rate max, SDR, and MMSE in terms of the transmission rate.

Table 2
The basic parameter settings.

Channel parameters		
C_0	The reference strength for the channel (dB)	−30
a_{ur}	fading factor from SD to RIS (Guo et al., 2023a)	2.8
a_{rb}	fading factor from RIS to BS (Guo et al., 2023a)	2
σ^2	Noise power (dBm)	−70
B	Bandwidth (MHz)	5
Active RIS parameters		
N	Number of reflecting elements	400
P_{dc}	Power consumed by control circuit (dBm)	−5
P_{sw}	Power consumed by phase shift switch (dBm)	−10
σ_r^2	The noise power introduced by RIS (dBm)	−70
Task parameters		
S	The basic unit of task size (Mb)	0.4
W	The basic unit of computing amount (GHz)	0.84
K_m	The multiples of the basic unit	[1, 20]
D_m^{Th}	The delay bound(ms)	10
Computing parameters		
M	Number of SDs	20
f^{M}	Computing resource of an SD (GHz)	1.2
f^{E}	Computing resource of MEC servers (GHz)	50
Energy parameters		
p^{Th}	The maximum transmit power (w)	0.1
β	The energy factor for local computing	0.25×10^{-18}
υ	The energy factor for MEC computing	10×10^{-28}

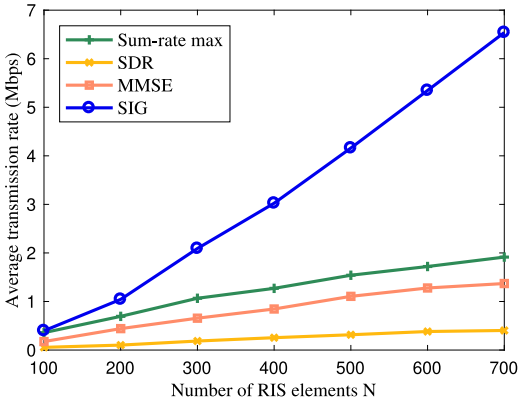

Fig. 3. RIS-assisted transmission rate improvement.

Sum-rate max (Guo et al., 2023b; Xu et al., 2022): Choosing the RIS's phase shift to maximize the sum of transmission rate at each iteration.

SDR (Wu and Zhang, 2019; Sun et al., 2022): Finding out the phase shift of the active RIS by semidefinite relaxation (SDR).

MMSE (Nadeem et al., 2019): The minimum mean square error-based channel estimation combined with a project gradient ascent method determines the optimum phase shift.

As illustrated in Fig. 3, the average transmission rate of all the investigated active RIS beamforming solvers increases with the number of RIS elements. The efficiency of our proposed SIG is demonstrated by giving the highest average transmission rate under various RIS elements.

Since additional steps with random values are needed to construct a rank-one solution from the obtained higher-rank solution under SDR in a multiple-user multiple-RIS element environment, the average transmission rate given by SDR is worse than other iterative beamforming methods, such as sum-rate max, MMSE and SIG, as demonstrated in Fig. 3.

The impact of amplification factor ρ on transmission power reduction is demonstrated in Fig. 4. With increasing ρ, the transmission power required by each SD to achieve the same transmission decreases, as shown in Fig. 4. Take SD $m = 5$ as an example; for achieving the

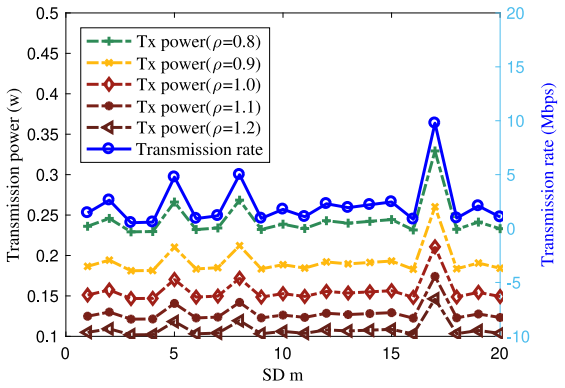

Fig. 4. The impact of ρ on transmission power reduction.

transmission rate of 4.8 Mbps, when the amplification factor $\rho = 0.8$, the user is required to transmit the offloading task with transmit power of 0.266 W; however, when $\rho = 1.2$, he needs to do that with a transmission power of 0.118 W. Note that the channel gain is different for each SD due to the dynamic nature of the wireless environment. Thus, each SD's achievable transmission rate differs from that of others. Therefore, the transmission rate varies among SDs, as illustrated in Fig. 4.

7.3. Evaluation of delay guarantee and energy efficiency

This subsection evaluates the efficiency of the proposed RISADA for the JCORO problem by comparing it with the following four benchmark schemes.

- *Local greedy:* Each task is computed locally on the local SD (Mao et al., 2021).
- Active RIS-assisted *MEC greedy:* All tasks are offloaded to the MEC server via the active RIS-assisted cellular network for edge computing (Yue et al., 2022). The active RIS's optimum phase shift and amplification factor are solved via the sum-rate max method described in Guo et al. (2023b), Xu et al. (2022). At the same time, the computing resources at the MEC servers are distributed among offloaded tasks through the RIS-assisted weighted fair (WFEC) policy described in Guo et al. (2023a).
- Active RIS-assisted *delay-constrained task partition and offloading (DTPO):* Making task partitioning and offloading decisions based on the delay requirements of tasks. In particular, for each SD, a task partition and offloading policy that minimizes its task latency is chosen (Yue et al., 2022; Guo et al., 2023b). Similar to *MEC greedy*, the active RIS's optimum phase shift and amplification factor are solved via sum-rate-max. At the same time, the computation resource allocation at the MEC server is addressed through WFEC.
- Active RIS-assisted *energy-efficient task partition and offloading (ETPO):* The task partition and offloading decisions are made for achieving energy efficiency while trying to reduce the latency of tasks, which is a variant of Mahenge et al. (2022). In addition, the active RIS's optimum phase shift and amplification factor are solved by the sum-rate max method.

Since the delay-based service satisfaction reflects the satisfaction level of the user to delay guarantee, we use percentage to evaluate it by setting $\zeta = \zeta \times 100\%$. Similarly, since the system utility reflects the integrated performance of delay guarantee and energy efficiency, we also use percentage to evaluate it by setting $G = \frac{G-G_{\min}}{G_{\max}-G_{\min}} \times 100\%$, where $G_{\min}$ and $G_{\max}$ are the minimum and maximum value of G in the simulation scenario.

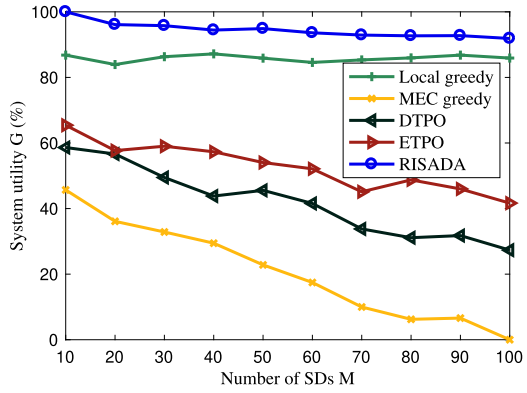

Fig. 5. System utility under various user numbers.

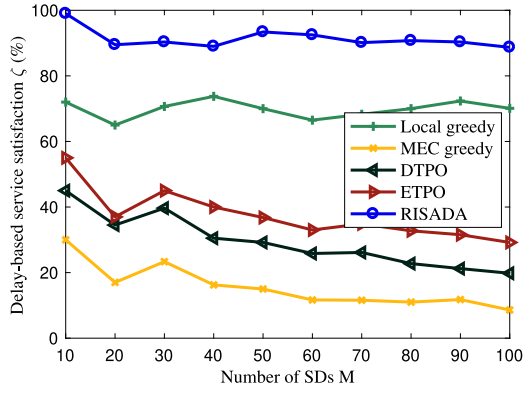

Fig. 6. Delay-based service satisfaction under various user numbers.

7.3.1. Adaptive to user number

First, we investigate the impact of the number of SDs on system utility and delay-based service satisfaction. As illustrated in Fig. 5, the system utilities given by MEC greedy, DTPO, ETPO, and RISADA decrease with the number of SDs. The reason is that as the number of SDs continues to increase, so do the computation tasks competing for shared RIS-assisted wireless and MEC computing resources. Thus, per-user service degrades in end-edge collaborative computation offloading schemes, such as MEC greedy, DTPO, ETPO, and RISADA. In other words, the delay-based service satisfaction given by the end-edge collaborative schemes decreases, as illustrated in Fig. 6, while energy consumption increases, as shown in Table 3. Therefore, it is unsurprising that the system utility (i.e., the weighted sum of delay-based service satisfaction and energy consumption) given by end-edge collaborative schemes, such as MEC greedy, DTPO, ETPO, and RISADA, decreases as SDs increase.

The efficiency of the proposed RISADA is demonstrated by always giving the highest system utility and delay-based service satisfaction as well as the lowest energy consumption compared to other investigated schemes, e.g., local greedy, MEC greedy, DTPO, and ETPO, as illustrated in Figs. 5–6 and Table 3, respectively. For example, when M reaches 100, the delay-based service satisfaction given by MEC greedy has reduced to about 10%. However, the delay-based service satisfaction given by RISADA could still be greater than 90%, as illustrated in Fig. 6.

Since all tasks are computed in SDs, the increase in the number of SDs has little impact on the local greedy scheme's offloading and resource allocation decisions. Therefore, as the number of SDs changes, local greedy gives roughly constant performance in terms of system utility, delay-based service satisfaction, and energy consumption, as illustrated in Figs. 5–6 and Table 3, respectively. The fluctuation of the performance is due to the varying task sizes.

Table 3
Energy consumption (J/task).

M	Local greedy	MEC greedy	DTPO	ETPO	RISADA
10	0.25	1.34	1.02	0.87	0.04
20	0.28	1.56	0.99	0.97	0.08
30	0.26	1.75	1.30	1.0	0.10
40	0.26	1.80	1.42	1.02	0.14
50	0.26	2.03	1.34	1.10	0.16
60	0.27	2.20	1.45	1.14	0.20
70	0.27	2.47	1.74	1.41	0.21
80	0.26	2.60	1.81	1.26	0.22
90	0.26	2.60	1.77	1.35	0.22
100	0.26	2.81	1.92	1.48	0.23

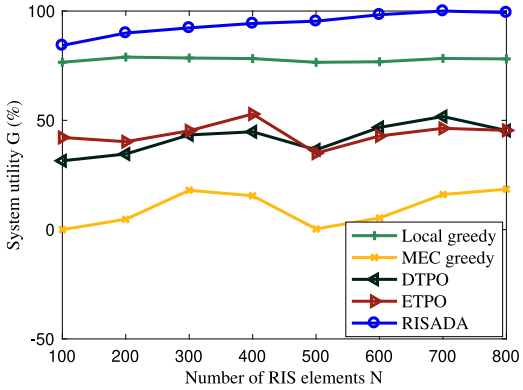

Fig. 7. System utility under various RIS element scale.

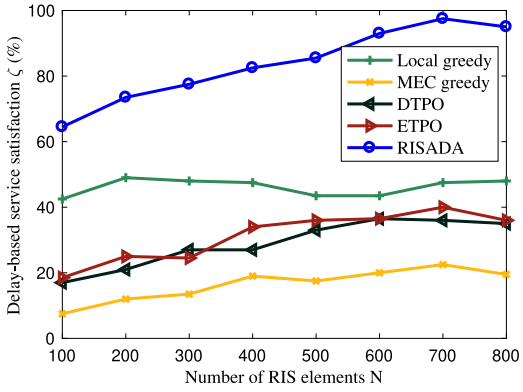

Fig. 8. Delay-based service satisfaction under various RIS element scales.

7.3.2. Adaptive to RIS scale

This subsection further observes the performance by varying the number of RIS elements. The system utility, delay-based service satisfaction, and energy consumption of the investigated schemes are shown in Figs. 7–8 and Table 4, respectively.

Since the transmission rate increases with the number of active RIS elements, it is not surprising that delay-based service satisfaction provided by end-edge collaborative schemes (e.g., MEC greedy, DTPO, ETPO and RISADA) increase with N, as illustrated in Fig. 8.

However, under the sum-rate max solver for phase shift and amplification factor of the active RIS, the energy consumptions given by MEC greedy, DTPO, and ETPO do not significantly decrease with N, as illustrated in Table 4. The reason is that, for pursuing transmission rate maximization, some SDs in the above schemes would transmit their tasks with their maximum available power. This leads to their energy consumption having little benefit from increasing N. Thus, the system utilities given by MEC greedy, DTPO, and ETPO do not increase significantly as N increases, as shown in Fig. 7.

Table 4
Energy consumption (J/task).

N	Local greedy	MEC greedy	DTPO	ETPO	RISADA
100	0.26	2.38	1.46	1.13	0.23
200	0.25	2.28	1.40	1.26	0.14
300	0.25	1.86	1.18	1.10	0.10
400	0.26	1.99	1.13	0.94	0.09
500	0.27	2.47	1.46	1.54	0.08
600	0.27	2.34	1.16	1.29	0.06
700	0.25	2.02	1.0	1.21	0.05
800	0.27	1.9	1.19	1.20	0.05

The effectiveness of the proposed RISADA is again validated by giving the highest delay-based service satisfaction (see Fig. 8) and lowest energy consumption (see Table 4), thus the highest system utility (see Fig. 7). This is because RISADA can dynamically adjust the transmission power and MEC computation resource allocation policy to adapt to the varying RIS scales and quality of wireless reflecting paths between offloading users and MEC servers to reduce energy consumption, considering the delay requirements of tasks. For example, when N reaches 800, the delay-based service satisfaction given by RISADA exceeds 90%, while that provided by MEC greedy only approximates 20%, as illustrated in Fig. 8. Meanwhile, the energy consumption reduces to 0.05 J/task in RISADA while that given by MEC greedy reaches 1.9 J/task, as shown in Table 4.

Since local computation is not affected by changes in N in the wireless transmission environment, the delay-based service satisfaction, energy consumption, and system utility given by local greedy is again stable under various N.

MEC greedy again performs the worst compared to other investigated schemes by giving the lowest system utility, delay-based service satisfaction, and highest energy consumption, as illustrated in Figs. 7–8 and Table 4, respectively. Although the increasing N could increase the transmission rate of the offloading tasks in MEC greedy, thus increasing the delay-based service satisfaction (see Fig. 8), the workload competing for the wireless transmission and MEC computation resources are still higher in MEC greedy (all tasks are offloaded) in comparison with distributed and end-edge collaborative computation schemes (e.g., local greedy, DTPO, ETPO and RISADA). Accordingly, MEC greedy gives the worst delay guarantee and energy consumption performance.

7.4. Convergence evaluation

This subsection investigates the convergence of RISADA. We observe the dynamics of the system utility of RISADA under various SD numbers. As shown in Fig. 9, the system utility under various M can converge to a stable value within a limited iteration. For example, when $M = 20$, the system utility reaches a constant after two iterations.

The number of iterations to converge increases with increasing M. For example, the number of iterations for convergence at $M = 20$, $M = 40$, $M = 60$ and $M = 80$ are 2, 10^-, 60^- and about 100, respectively where X^- means less than X, which is accordance with our intuition. This is because the competition for the wireless reflecting path and computation resources in computation offloading schemes increases with the increasing SD number.

8. Conclusion

This paper has proposed an active RIS-assisted end-edge collaborative task partitioning and offloading scheme, termed RISADA, to address the joint computation offloading and active RIS optimization problem in industrial edge computing to achieve a low-delay guarantee while reducing energy consumption. In particular, the task partitioning and offloading properties, considering the delay requirements and energy consumption, have been theoretically analyzed. The joint active RIS optimum and MEC computation resource allocation properties have also been theoretically investigated. Based on the analytical results, a RISADA scheme was proposed to solve this problem. The simulation results have shown our scheme's performance advantages over the benchmark schemes regarding delay guarantee and energy efficiency.

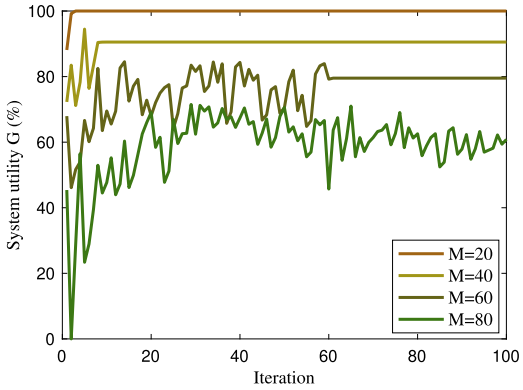

Fig. 9. Convergence rate.

CRediT authorship contribution statement

Mian Guo: Writing – original draft, Visualization, Validation, Software, Data curation. **Yuehong Chen:** Writing – review & editing, Writing – original draft, Visualization, Validation, Investigation, Data curation. **Zhiping Peng:** Writing – review & editing, Supervision, Conceptualization. **Qirui Li:** Writing – review & editing, Data curation. **Keqin Li:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China [grant number 62273109].

Appendix

A.1. Proof of Theorem 1

Proof. For delay guarantee, $D_m \le D_m^{\mathrm{Th}}$ should hold, thus we have

$$\begin{cases} D_m^{\mathrm{L}} \le D_m^{\mathrm{Th}}, \\ D_m^{\mathrm{O}} \le D_m^{\mathrm{Th}}. \end{cases}$$

Substituting (9) and (23) into the above formula, and with some calculus, we have

$$\begin{cases} \alpha_m \ge 1 - \dfrac{D_m^{\mathrm{Th}} f_m^{\mathrm{M}}}{K_m W}, \\ \alpha_m \le \dfrac{D_m^{\mathrm{Th}}}{\psi_m K_m W}. \end{cases}$$

which ends the proof.

A.2. Proof of Lemma 1

Proof. If $D_m = D_m^{\mathrm{L}}$, then we have $D_m^{\mathrm{L}} \geq D_m^{\mathrm{O}}$. According to (24), we have $(1-\alpha_m)/f_m^{\mathrm{M}} \geq \alpha_m \psi_m$, that is, we have

$$\alpha_m \leq \frac{1}{1+\psi_m f_m^{\mathrm{M}}}. \tag{54}$$

Combining (54) and (25), we have

$$\begin{cases} \alpha_m^* \geq \alpha_m^{\mathrm{lb\_l}}, & (55) \\ \alpha_m^* \leq \alpha_m^{\mathrm{ub\_l}}, & (56) \end{cases}$$

where $\alpha_m^{\mathrm{lb\_l}} \triangleq 1 - \frac{D_m^{\mathrm{Th}} f_m^{\mathrm{M}}}{K_m W}$, and $\alpha_m^{\mathrm{ub\_l}} \triangleq \min\left(\frac{D_m^{\mathrm{Th}}}{\psi_m K_m W}, \frac{1}{1+\psi_m f_m^{\mathrm{M}}}\right)$.

According to (16), if $\omega_m \geq \beta f_m^{\mathrm{M}}$, then when $\alpha_m^* = \alpha_m^{\mathrm{lb\_l}}$, we could obtain the lower bound of the delay guaranteed energy consumption, that is

$$E_m^* \geq K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right).$$

However, if $\omega_m < \beta f_m^{\mathrm{M}}$ holds, the lower bound of the delay guaranteed energy consumption is yielded when $\alpha_m^* = \alpha_m^{\mathrm{ub\_l}}$, that is

$$E_m^* \geq K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{ub\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right).$$

Therefore, we have

$$E_m^* \geq \begin{cases} K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \omega_m \geq \beta f_m^{\mathrm{M}} \\ K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{ub\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \text{otherwise.} \end{cases} \tag{57}$$

Substituting (56) into (17), and replacing D_m with D_m^{L}, we have

$$\begin{aligned} \zeta_m^* &= \mathbf{1}(D_m^* \leq D_m^{\mathrm{Th}}) \\ &= \mathbf{1}(D_m^* \leq D_m^{\mathrm{Th}}) \\ &= \mathbf{1}\left(\frac{(1-\alpha_m)K_m W}{f_m^{\mathrm{M}}} \leq D_m^{\mathrm{Th}}\right) \\ &\leq \mathbf{1}\left(\frac{(1-\alpha_m^{\mathrm{ub\_l}})K_m W}{f_m^{\mathrm{M}}} \leq D_m^{\mathrm{Th}}\right). \end{aligned} \tag{58}$$

According to (18), we have

$$G_m^* = \varphi \zeta_m^* - (1-\varphi) E_m^* / E^{\max}. \tag{59}$$

Let $\varphi_2 = \frac{(1-\varphi)}{E^{\max}}$, and substituting (57) and (58) into (59), then

$$\begin{aligned} G_m^* =& \varphi \zeta_m^* - \varphi_2 E_m^* \\ \leq& \varphi \mathbf{1}\left(\frac{(1-\alpha_m^{\mathrm{ub\_l}})K_m W}{f_m^{\mathrm{M}}} \leq D_m^{\mathrm{Th}}\right) \\ & - \varphi_2 K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right). \end{aligned}$$

Since $\mathbf{1}\left(\frac{(1-\alpha_m^{\mathrm{lb\_l}})K_m W}{f_m^{\mathrm{M}}} \leq D_m^{\mathrm{Th}}\right) = \mathbf{1}\left(\frac{(1-\alpha_m^{\mathrm{ub\_l}})K_m W}{f_m^{\mathrm{M}}} \leq D_m^{\mathrm{Th}}\right) = 1$ holds, we have

$$G_m^* \leq \begin{cases} \varphi - \varphi_2 K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \omega_m \geq \beta f_m^{\mathrm{M}} \\ \varphi - \varphi_2 K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{ub\_l}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \text{otherwise.} \end{cases}$$

Then the statement follows.

A.3. Proof of Lemma 2

Proof. If $D_m = D_m^{\mathrm{O}}$, then we have $D_m^{\mathrm{L}} \leq D_m^{\mathrm{O}}$. According to (24), we have $(1-\alpha_m)/f_m^{\mathrm{M}} \leq \alpha_m \psi_m$, that is, we have

$$\alpha_m \geq \frac{1}{1+\psi_m f_m^{\mathrm{M}}}. \tag{60}$$

Combining (25) and (60), we have

$$\begin{cases} \alpha_m^* \geq \alpha_m^{\mathrm{lb\_o}}, & (61) \\ \alpha_m^* \leq \alpha_m^{\mathrm{ub\_o}}, & (62) \end{cases}$$

where $\alpha_m^{\mathrm{lb\_o}} \triangleq \max\left(1 - \frac{D_m^{\mathrm{Th}} f_m^{\mathrm{M}}}{K_m W}, \frac{1}{1+\psi_m f_m^{\mathrm{M}}}\right)$, and $\alpha_m^{\mathrm{ub\_o}} \triangleq \frac{D_m^{\mathrm{Th}}}{\psi_m K_m W}$.

According to (16), if $\omega_m \geq \beta f_m^{\mathrm{M}}$, then when $\alpha_m^* = \alpha_m^{\mathrm{lb\_o}}$, we could obtain the lower bound of the delay guaranteed energy consumption, that is

$$E_m^* \geq K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_o}}(\omega_m - \beta f_m^{\mathrm{M}})\right).$$

However, if $\omega_m < \beta f_m^{\mathrm{M}}$ holds, the lower bound of the delay guaranteed energy consumption is yielded when $\alpha_m^* = \alpha_m^{\mathrm{ub\_o}}$, that is

$$E_m^* \geq K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{ub\_o}}(\omega_m - \beta f_m^{\mathrm{M}})\right).$$

Therefore, we have

$$E_m^* \geq \begin{cases} K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_o}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \omega_m \geq \beta f_m^{\mathrm{M}} \\ K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{ub\_o}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \text{otherwise.} \end{cases} \tag{63}$$

Substituting (61) into (17), and replacing D_m with D_m^{O}, we have

$$\begin{aligned} \zeta_m^* &= \mathbf{1}(D_m^{\mathrm{O}} \leq D_m^{\mathrm{Th}}) \\ &= \mathbf{1}(\alpha_m K_m W \psi_m \leq D_m^{\mathrm{Th}}) \\ &\leq \mathbf{1}(\alpha_m^{\mathrm{ub\_o}} K_m W \psi_m \leq D_m^{\mathrm{Th}}). \end{aligned} \tag{64}$$

Since $\mathbf{1}(\alpha_m^{\mathrm{lb\_o}} K_m W \psi_m \leq D_m^{\mathrm{Th}}) = \mathbf{1}(\alpha_m^{\mathrm{ub\_o}} K_m W \psi_m \leq D_m^{\mathrm{Th}}) = 1$ holds, substituting (63) and (64) into (18), we have

$$G_m^* \leq \begin{cases} \varphi - \varphi_2 K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_o}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \omega_m \geq \beta f_m^{\mathrm{M}} \\ \varphi - \varphi_2 K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{ub\_o}}(\omega_m - \beta f_m^{\mathrm{M}})\right), \text{otherwise.} \end{cases}$$

Then the statement follows.

A.4. Proof of Theorem 2

Proof. Since $D_m \geq D_m^{\mathrm{Th}}$, according to (17), we have $\zeta_m = 0$, thus, according to (18), we have

$$G_m = -(1-\varphi) E_m / E^{\max}, \tag{65}$$

which monotonically decreases with E_m.

Furthermore, based on (16), we have

Case a: If $\omega_m \geq \beta f_m^{\mathrm{M}}$, then E_m monotonically increases with α_m, accordingly, $\alpha_m^* = 0$ is a dominated offloading decision, which minimizes E_m thus maximizes G_m.

Case b:If $\omega_m \leq \beta f_m^{\mathrm{M}}$, then E_m monotonically decreases with α_m, accordingly, $\alpha_m^* = 1$ is a dominated offloading decision, which minimizes E_m thus maximizes G_m.

Therefore, substituting α_m^* into (16), we yield $E_m^* = K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^*(\omega_m - \beta f_m^{\mathrm{M}})\right)$, and further substituting E_m^* into (65), we have $G_m^* = -(1-\varphi) E_m^* / E^{\max}$.

Then the statement follows.

A.5. Proof of Theorem 3

Proof. Under any delay-guaranteed RIS optimum and MEC resource allocation policy, we have $D_m^{\mathrm{O}} \leq D_m^{\mathrm{Th}}$, substituting into (23), we have $\alpha_m K_m W (\frac{\chi}{R_m} + \frac{1}{f_m^{\mathrm{E}}}) \leq D_m^{\mathrm{Th}}$, accordingly, we have

$$R_m \geq \frac{\chi}{\frac{D_m^{\mathrm{Th}}}{\alpha_m K_m W} - \frac{1}{f_m^{\mathrm{E}}}}.$$

Substituting R_m into (3) and further combining with (2), we have

$$B \log_2\left(1 + \frac{p_m |\mathbf{g}\boldsymbol{\Phi} h_m|^2}{\sigma_r^2 \|\mathbf{g}\boldsymbol{\Phi}\|^2 + \sigma^2/\rho^2}\right) \geq \frac{\chi}{\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{1}{f_m^{\mathrm{E}}}}$$

$$\Longrightarrow p_m \geq \left(\ln\left(\frac{\chi}{B(\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{1}{f_m^{\mathrm{E}}})}\right) - 1\right) \frac{\sigma_r^2 \|\mathbf{g}\boldsymbol{\Phi}\|^2 + \sigma^2/\rho^2}{|\mathbf{g}\boldsymbol{\Phi} h_m|^2}.$$

Accordingly, for delay constraints, the transmission power should satisfy

$$\begin{cases} p_m \geq \left(\ln \left(\dfrac{\chi}{B(\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{1}{f_m^{\mathrm{E}}})} \right) - 1 \right) \dfrac{\sigma_r^2 \|\mathbf{g}\Phi\|^2 + \sigma^2/\rho^2}{|\mathbf{g}\Phi h_m|^2}, \\ p_m \leq p^{\mathrm{Th}}. \end{cases}$$

The statement then follows.

A.6. Proof of Theorem 4

Proof. Under any delay-guaranteed RIS optimum and MEC resource allocation policy, we have $D_m^{\mathrm{O}} \leq D_m^{\mathrm{Th}}$, substituting into (23), we have $\alpha_m K_m W (\frac{\chi}{R_m} + \frac{1}{f_m^{\mathrm{E}}}) \leq D_m^{\mathrm{Th}}$, accordingly, we have

$$f_m^{\mathrm{E}} \geq \frac{1}{\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{\chi}{R_m}}.$$

On the other hand, according to the computation resource constraints of the MEC server (see (20f)), the computation resource allocated to offloaded subtasks from the mth IoT user is upper bounded by $\sum_{m \in \mathcal{M}} f_m^{\mathrm{E}} \leq f^{\mathrm{E}}$.

Therefore, for satisfying the delay requirement, the computation resource allocation at MEC server for the subtasks from the mth ($\forall m \in \mathcal{M}$) IoT user should satisfy

$$\begin{cases} f_m^{\mathrm{E}} \geq \dfrac{1}{\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{\chi}{R_m}}, \\ \sum\limits_{m \in \mathcal{M}} f_m^{\mathrm{E}} \leq f^{\mathrm{E}}. \end{cases}$$

Then the statement follows.

A.7. Proof of Lemma 3

Proof. According to (31), we have

$$G_m^* \leq \varphi \frac{\log\left(1 + \max(0, (D_m^{\mathrm{Th}} - \alpha_m^{\mathrm{ub\_o}} K_m W \psi_m))\right)}{\log(1 + D_m^{\mathrm{Th}})} - \varphi_2 K_m W \left(\beta f_m^{\mathrm{M}} + \alpha_m^{\mathrm{lb\_o}} (\omega_m - \beta f_m^{\mathrm{M}})\right). \tag{66}$$

Given α_m, R_m, since $\psi_m = \frac{\chi}{R_m} + \frac{1}{f_m^{\mathrm{E}}}$, $\omega_m = (p_m + Q^{\mathrm{RIS}})\chi / R_m + \upsilon f_m^{\mathrm{E}}$, ζ_m monitonically increases with f_m^{E}, ω_m increases with f_m^{E}.

Case I: If $\omega_m \geq \beta f_m^{\mathrm{M}}$, then according to the definition of ω_m, we have

$$f_m^{\mathrm{E}} \geq \frac{1}{\upsilon}(\beta f_m^{\mathrm{M}} - (p_m + Q^{\mathrm{RIS}})\chi / R_m). \tag{67}$$

In this case, E_m increases with f_m^{E}, thus G_m decreases with f_m^{E}. Therefore, when $f_m^{\mathrm{E}*}$ satisfies

$$\begin{cases} f_m^{\mathrm{E}*} \geq \max(f_m^{\mathrm{lb\_a}}, f_m^{\mathrm{lb\_b}}) \\ f_m^{\mathrm{E}*} + \sum\limits_{m' \in \mathcal{M} \setminus \{m\}} f_{m'}^{\mathrm{E}} \leq f^{\mathrm{E}}. \end{cases}$$

where $f_m^{\mathrm{lb\_a}} = \frac{1}{\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{\chi}{R_m}}$, $f_m^{\mathrm{lb\_b}} = \frac{1}{\upsilon}(\beta f_m^{\mathrm{M}} - (p_m + Q^{\mathrm{RIS}})\chi / R_m)$, we get the optimum G_m^*.

Case II: If $\omega_m \leq \beta f_m^{\mathrm{M}}$, then according to the definition of ω_m, we have

$$f_m^{\mathrm{E}} \leq \frac{1}{\upsilon}(\beta f_m^{\mathrm{M}} - (p_m + Q^{\mathrm{RIS}})\chi / R_m). \tag{68}$$

In this case, E_m decreases with f_m^{E}, thus G_m increases with f_m^{E}. Therefore, when $f_m^{\mathrm{E}*}$ satisfies

$$\begin{cases} f_m^{\mathrm{E}*} \geq \dfrac{1}{\frac{D_m^{\mathrm{Th}}}{K_m W \alpha_m} - \frac{\chi}{R_m}} \\ f_m^{\mathrm{E}*} \leq \min(f_m^{\mathrm{lb\_b}}, f_m^{\mathrm{ub}}). \end{cases}$$

where $f_m^{\mathrm{ub}} = f^{\mathrm{E}} - \sum_{m' \in \mathcal{M} \setminus \{m\}} f_{m'}^{\mathrm{E}}$.

Then the statement follows.

Data availability

Data will be made available on request.

References

Akhlaqi, M.Y., Mohd Hanapi, Z.B., 2023. Task offloading paradigm in mobile edge computing-current issues, adopted approaches, and future directions. J. Netw. Comput. Appl. 212, 103568.

Chen, Q., Guo, S., Wang, K., Xu, W., Li, J., Cai, Z., Gao, H., Zomaya, A., 2023. Towards real-time inference offloading with distributed edge computing: the framework and algorithms. IEEE Trans. Mob. Comput..

Chen, Y., Liu, Z., Zhang, Y., Wu, Y., Chen, X., Zhao, L., 2021. Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial internet of things. IEEE Trans. Ind. Inform. 17 (7), 4925–4934.

Chen, Y., Zhao, J., Hu, J., Wan, S., Huang, J., 2024. Distributed task offloading and resource purchasing in NOMA-enabled mobile edge computing: Hierarchical game theoretical approaches. ACM Trans. Embed. Comput. Syst. 23 (1), 1–28.

Dai, X., Xiao, Z., Jiang, H., Alazab, M., Lui, J.C.S., Dustdar, S., Liu, J., 2023. Task co-offloading for D2D-assisted mobile edge computing in industrial internet of things. IEEE Trans. Ind. Inform. 19 (1), 480–490.

Deng, X., Yin, J., Guan, P., Xiong, N.N., Zhang, L., Mumtaz, S., 2023. Intelligent delay-aware partial computing task offloading for multiuser industrial internet of things through edge computing. IEEE Internet Things J. 10 (4), 2954–2966.

Guo, M., Mukherjee, M., Lloret, J., 2023a. RIS-assisted edge-D2D cooperative edge computing for industrial applications. Comput. Commun. 206, 178–188.

Guo, M., Xu, C., Mukherjee, M., 2023b. RIS-assisted device-edge collaborative edge computing for industrial applications. Peer- To- Peer Netw. Appl. 16 (5), 2023–2038.

Habiba, U., Maghsudi, S., Hossain, E., 2024. A repeated auction model for load-aware dynamic resource allocation in multi-access edge computing. IEEE Trans. Mob. Comput. 23 (7), 7801–7817.

Han, G., Xu, Z., Zhu, H., Ge, Y., Peng, J., 2024. A two-stage model based on a complex-valued separate residual network for cross-domain IIoT devices identification. IEEE Trans. Ind. Inform. 20 (2), 2589–2599.

Jia, Y., Liu, B., Zhang, X., Dai, F., Khan, A., Qi, L., Dou, W., 2024. Model pruning-enabled federated split learning for resource-constrained devices in artificial intelligence empowered edge computing environment. ACM Trans. Sen. Netw. Just Accepted.

Leng, J., Sha, W., Wang, B., Zheng, P., Zhuang, C., Liu, Q., Wuest, T., Mourtzis, D., Wang, L., 2022. Industry 5.0: Prospect and retrospect. J. Manuf. Syst. 65, 279–295.

Lin, Z., Qu, G., Chen, X., Huang, K., 2024. Split learning in 6G edge networks. IEEE Wirel. Commun. 31 (4), 170–176.

Liu, Z., Li, Z., Gong, Y., Wu, Y.-C., 2024. RIS-aided cooperative mobile edge computing: Computation efficiency maximization via joint uplink and downlink resource allocation. IEEE Trans. Wirel. Commun. 23 (9), 11535–11550.

Lv, L., Luo, H., Yang, L., Ding, Z., Nallanathan, A., Al-Dhahir, N., Chen, J., 2024. RIS-assisted wireless powered MEC: Multiple access design and resource allocation. IEEE Trans. Wirel. Commun. 1–1.

Mahenge, M.P.J., Li, C., Sanga, C.A., 2022. Energy-efficient task offloading strategy in mobile edge computing for resource-intensive mobile applications. Digit. Commun. Netw. 8 (6), 1048–1058.

Mao, S., Chu, X., Wu, Q., Liu, L., Feng, J., 2021. Intelligent reflecting surface enhanced D2D cooperative computing. IEEE Wirel. Commun. Lett. 10 (7), 1419–1423.

Nadeem, Q.-U.-A., Kammoun, A., Chaaban, A., Debbah, M., Alouini, M.-S., 2019. Intelligent reflecting surface assisted wireless communication: Modeling and channel estimation. arXiv preprint arXiv:1906.02360.

Navarro-Ortiz, J., Romero-Diaz, P., Sendra, S., Ameigeiras, P., Ramos-Munoz, J.J., Lopez-Soler, J.M., 2020. A survey on 5G usage scenarios and traffic models. IEEE Commun. Surv. Tutorials 22 (2), 905–929.

Peng, P., Lin, W., Wu, W., Zhang, H., Peng, S., Wu, Q., Li, K., 2024. A survey on computation offloading in edge systems: From the perspective of deep reinforcement learning approaches. Comput. Sci. Rev. 53, 100656.

Peng, K., Xiao, P., Wang, S., Leung, V.C.M., 2023. AoI-aware partial computation offloading in IIoT with edge computing: A deep reinforcement learning based approach. IEEE Trans. Cloud Comput. 11 (4), 3766–3777.

Qian, L., Wu, Y., Jiang, F., Yu, N., Lu, W., Lin, B., 2021. NOMA assisted multi-task multi-access mobile edge computing via deep reinforcement learning for industrial internet of things. IEEE Trans. Ind. Inform. 17 (8), 5688–5698.

Shi, Z., Lu, H., Xie, X., Yang, H., Huang, C., Cai, J., Ding, Z., 2023. Active RIS-aided EH-NOMA networks: A deep reinforcement learning approach. IEEE Trans. Commun. 71 (10), 5846–5861.

Songhorabadi, M., Rahimi, M., MoghadamFarid, A., Haghi Kashani, M., 2023. Fog computing approaches in IoT-enabled smart cities. J. Netw. Comput. Appl. 211, 103557.

Su, Y., Fan, W., Liu, Y., Wu, F., 2023. A truthful combinatorial auction mechanism towards mobile edge computing in industrial internet of things. IEEE Trans. Cloud Comput. 11 (2), 1678–1691.
Sun, C., Ni, W., Bu, Z., Wang, X., 2022. Energy minimization for intelligent reflecting surface-assisted mobile edge computing. IEEE Trans. Wirel. Commun. 21 (8), 6329–6344.
Tan, L., Guo, S., Zhou, P., Kuang, Z., Jiao, X., 2024. HAT: Task offloading and resource allocation in RIS-assisted collaborative edge computing. IEEE Trans. Netw. Sci. Eng. 11 (5), 4665–4678.
Wei, X., Shen, D., Dai, L., 2021a. Channel estimation for RIS assisted wireless communications—Part I: Fundamentals, solutions, and future opportunities. IEEE Commun. Lett. 25 (5), 1398–1402.
Wei, X., Shen, D., Dai, L., 2021b. Channel estimation for RIS assisted wireless communications—Part II: An improved solution based on double-structured sparsity. IEEE Commun. Lett. 25 (5), 1403–1407.
Wu, Q., Zhang, R., 2019. Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming. IEEE Trans. Wirel. Commun. 18 (11), 5394–5409.
Xie, H., Li, D., Gu, B., 2024. Exploring hybrid active-passive RIS-aided MEC systems: From the mode-switching perspective. IEEE Trans. Wirel. Commun. 1–1.
Xu, F., Ye, Z., Cao, H., Hu, Z., 2022. Sum-rate optimization for IRS-aided D2D communication underlaying cellular networks. IEEE Access 10, 48499–48509.
Yu, J., Li, Y., Liu, X., Sun, B., Wu, Y., Hin-Kwok Tsang, D., 2023. IRS assisted NOMA aided mobile edge computing with queue stability: Heterogeneous multi-agent reinforcement learning. IEEE Trans. Wirel. Commun. 22 (7), 4296–4312.
Yue, S., Ren, J., Qiao, N., Zhang, Y., Jiang, H., Zhang, Y., Yang, Y., 2022. TODG: Distributed task offloading with delay guarantees for edge computing. IEEE Trans. Parallel Distrib. Syst. 33 (7), 1650–1665.
Zeng, C., Wang, X., Zeng, R., Li, Y., Shi, J., Huang, M., 2024. Joint optimization of multi-dimensional resource allocation and task offloading for QoE enhancement in cloud-edge-end collaboration. Future Gener. Comput. Syst. 155, 121–131.
Zhang, Z., Dai, L., Chen, X., Liu, C., Yang, F., Schober, R., Poor, H.V., 2022. Active RIS vs. passive RIS: Which will prevail in 6G? IEEE Trans. Commun. 71 (3), 1707–1725.
Zhang, F., Han, G., Liu, L., Zhang, Y., Peng, Y., Li, C., 2024a. Cooperative partial task offloading and resource allocation for IIoT based on decentralized multiagent deep reinforcement learning. IEEE Internet Things J. 11 (3), 5526–5544.
Zhang, T., Xu, D., Tolba, A., Yu, K., Song, H., Yu, S., 2024b. Reinforcement-learning-based offloading for RIS-aided cloud–edge computing in IoT networks: Modeling, analysis, and optimization. IEEE Internet Things J. 11 (11), 19421–19439.
Zhao, M., Liu, C., Zhu, S., 2025. Joint optimization scheme for task offloading and resource allocation based on MO-MFEA algorithm in intelligent transportation scenarios. J. Netw. Comput. Appl. 233, 104039.
Zheng, B., You, C., Mei, W., Zhang, R., 2022. A survey on channel estimation and practical passive beamforming design for intelligent reflecting surface aided wireless communications. IEEE Commun. Surv. Tutorials 24 (2), 1035–1071.
Zhi, K., Pan, C., Ren, H., Chai, K.K., Elkashlan, M., 2022. Active RIS versus passive RIS: Which is superior with the same power budget? IEEE Commun. Lett. 26 (5), 1150–1154.
Zhou, J., Hou, X., Zeng, Y., Cong, P., Jiang, W., Guo, S., 2025. Quality of experience and reliability-aware task offloading and scheduling for multi-user mobile-edge computing systems. IEEE Trans. Serv. Comput. 1–14.
Zhou, F., You, C., Zhang, R., 2020. Delay-optimal scheduling for IRS-aided mobile edge computing. IEEE Wirel. Commun. Lett. 10 (4), 740–744.
Zhu, Y., Mao, B., Kato, N., 2022. A dynamic task scheduling strategy for multi-access edge computing in IRS-aided vehicular networks. IEEE Trans. Emerg. Top. Comput. 10 (4), 1761–1771.

Mian Guo received her Ph.D. in communication and information systems from South China University of Technology, China, in 2012. She was a visiting professor with the University of Ottawa, Ottawa, Canada, in 2016, and a visiting professor with Beihang University, China, in 2017. She is currently an associate professor at Guangdong Polytechnic Normal University, China. Her research interests include resource allocation, QoS provisioning in computer and communication networks, edge computing, and deep reinforcement learning.

Yuehong Chen received her master's degree in Applied mathematics major of Science College of Nanchang University, Nanchang, in 2004. She is an associate professor of School of mathematics of Guangdong Polytechnic Normal University. Her research interests include qualitative and stability studies of differential equations, as well as AI.

Zhiping Peng received his Ph.D. degree in Computer Science from South China University of Technology in 2007. He is currently a professor at Jiangmen Polytechnic, China. His research interests include cloud computing, resource allocation, and deep reinforcement learning.

Qirui Li received his Ph.D. degree from Guangzhou University (GZHU), Guangzhou, China. He is a professor at Guangdong University of Petrochemical Technology (GDUPT), Maoming, China. His research interests include wireless communication, artificial intelligence, cloud computing, image processing, and pattern recognition.

Keqin Li Keqin Li received a B.S. degree in computer science from Tsinghua University in 1985 and a Ph.D. degree in computer science from the University of Houston in 1990. He is a SUNY Distinguished Professor with the State University of New York and a National Distinguished Professor with Hunan University (China). He has authored or co-authored more than 1000 journal articles, book chapters, and refereed conference papers. He received several best paper awards from international conferences including PDPTA-1996, NAECON-1997, IPDPS-2000, ISPA-2016, NPC-2019, ISPA-2019, and CPSCom-2022. He holds nearly 75 patents announced or authorized by the Chinese National Intellectual Property Administration. He is among the world's top five most influential scientists in parallel and distributed computing in terms of single-year and career-long impacts based on a composite indicator of the Scopus citation database. He was a 2017 recipient of the Albert Nelson Marquis Lifetime Achievement Award for being listed in Marquis *Who's Who in Science and Engineering, Who's Who in America, Who's Who in the World, and Who's Who in American Education* for over twenty consecutive years. He received the Distinguished Alumnus Award from the Computer Science Department at the University of Houston in 2018. He received the IEEE TCCLD *Research Impact Award* from the IEEE CS Technical Committee on Cloud Computing in 2022 and the IEEE TCSVC *Research Innovation Award* from the IEEE CS Technical Community on Services Computing in 2023. He won the IEEE Region 1 *Technological Innovation Award* (Academic) in 2023. He is a Member of the SUNY Distinguished Academy. He is an AAAS Fellow, an IEEE Fellow, an AAIA Fellow, and an ACIS Founding Fellow. He is an Academician Member and Fellow of the International Artificial Intelligence Industry Alliance. He is a Member of Academia Europaea (Academician of the Academy of Europe).