Aerial-IRSs-Assisted Energy-Efficient Task Offloading and Computing

Wenwen Jiang, *Graduate Student Member, IEEE*, Bo Ai, *Fellow, IEEE*, Mushu Li, *Member, IEEE*,
Wen Wu, *Senior Member, IEEE*, Yingying Pei, *Graduate Student Member, IEEE*,
and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Timely and energy-efficient task offloading and computing can be challenging in mobile edge computing (MEC) networks when the communication links between devices and edge servers are unreliable. In this article, we apply multiple aerial intelligent reflective surfaces (AIRSs) to assist devices in offloading computing tasks to the edge server in a timely and reliable manner in the MEC network with poor offloading environments. To evaluate the timeliness of offloading and computing, we derive the evolution process of Age-of-Information (AoI) under the random arrival of the computing tasks. The association between devices and AIRSs, offloading order of computing tasks, design of IRS phase shift, and allocation of communication and computing resources are jointly optimized to minimize the average AoI and system energy consumption given computing requirements. To solve the formulated minimization problem, we propose an efficient problem-solving framework to cope with the challenge of variable coupling. First, we derive a closed-form optimal IRS phase shift to provide a reliable offloading environment. Then, we optimize the association between devices and AIRSs while reducing the offloading complexity and balancing the number of devices associated with each AIRS. Finally, we develop a low-complexity task offloading and resource allocation algorithm based on convex optimization to attain a good enough solution. Simulation results indicate the proposed solution outperforms benchmarks in timeliness and energy saving.**

***Index Terms*—Aerial intelligent reflective surface (AIRS), Age of Information (AoI), energy consumption, mobile edge computing (MEC).**

I. Introduction

THE BOOMING advancement of the Internet of Things (IoT) [1] has spawned many new applications, e.g., virtual/augmented reality [2], autonomous driving [3], and intelligent factories [4]. These burgeoning applications impose higher requirements on reliability, timeliness, and energy consumption, which need to be supported by providing sufficient resources. However, as devices move toward lightweight, shrinking device size limits the size of built-in batteries and computing servers. The devices of small size and low-computing capability are resource-constrained and cannot accommodate time-critical applications that require low-latency computing. Although the remote cloud can provide reliable service for such devices, the excessive offloading delay and energy consumption caused by long distances between devices and cloud servers are usually unbearable. Fortunately, mobile edge computing (MEC) [5] has constituted a promising solution for solving the above issues by providing cloud-like computing service at the edge of networks, which facilitates the timely execution of computing extensive tasks. The computing tasks generated by devices can be offloaded to the hybrid access point (HAP) integrated with the MEC server for execution while achieving high-resource utilization and low latency.

Nonetheless, the offloading and computing of tasks generated by resource-constrained devices at the MEC networks still face several inevitable challenges, whether for partial or binary offloading. First, wireless channels may suffer from severe attenuation caused by long communication distances or blockage of obstacles [6], which results in unreliable and outdated offloading, often invalidating the information contained in the computing results. Second, it is challenging to offload the computing tasks generated by multiple devices simultaneously due to the limited physical resources and energy budgets. An efficient scheduling scheme for offloading is thus necessary. Additionally, the unpredictable arrival of computing tasks makes the design of a scheduling scheme for task offloading more intractable. Third, efficient resource allocation is essential to achieve timely, reliable, and energy-saving offloading and computing, particularly in scenarios

Manuscript received 11 November 2023; revised 28 January 2024; accepted 19 February 2024. Date of publication 29 February 2024; date of current version 23 May 2024. This work was supported in part by the National Key Research and Development Program under Grant 2021YFB2900301, Grant 2020YFB1806604, and Grant 2021YFB3901302; in part by the NSFC under Grant 62221001 and Grant 62201311; in part by the Fundamental Research Funds for the Central Universities under Grant 2022JBQY004; in part by the Royal Society Newton Advanced Fellowship under Grant NA191006 and Grant 61961130391; in part by the Natural Science Foundation of Jiangsu Province Major Project under Grant BK20212002; in part by the Natural Sciences and Engineering Research Council of Canada; and in part by the Peng Cheng Laboratory Major Key Project under Grant PCL2023AS1-5 and Grant PCL2021A09-B2. *(Corresponding author: Bo Ai.)*

Wenwen Jiang and Bo Ai are with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China (e-mail: wenwenjiang@bjtu.edu.cn; boai@bjtu.edu.cn).

Mushu Li is with the Department of Electrical, Computer, and Biomedical Engineering, Toronto Metropolitan University, Toronto, ON M5B 2K3, Canada (e-mail: mushu.li@ryerson.ca).

Wen Wu is with the Frontier Research Center, Peng Cheng Laboratory, Shenzhen 518055, China (e-mail: wuw02@pcl.ac.cn).

Yingying Pei and Xuemin Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: y32pei@uwaterloo.ca; sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/JIOT.2024.3371586

2327-4662 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

with limited resources. Furthermore, analyzing the coupling relations between communication and computing resources for different scenarios presents a significant challenge.

To overcome the challenges, we deploy multiple aerial intelligent reflective surfaces (AIRSs) as passive relays to facilitate offloading in MEC networks, especially in cases with weak direct links between devices and the HAP. As one of the emerging technologies in 6G [7], intelligent reflective surface (IRS) [8] can intelligently reconfigure the signal propagation environment by adjusting the passive reflection elements with special physical structures. AIRS [9], [10], [11] is equipped with IRS on a high-altitude platform, such as an unmanned aerial vehicle (UAV) or high-altitude balloon, to increase the flexibility of IRS deployment and achieve the purpose of enhancing channel quality. AIRS inherits the advantages of UAV and IRS, enabling it to establish a high-probability Line-of-Sight (LoS) link with ground equipment [12] and providing more design freedom than terrestrial IRS [13], [14]. Consequently, the timeliness and reliability of offloading can be potentially guaranteed with the assistance of AIRSs, especially in scenarios with limited available resources and poor offloading environments.

To depict the timeliness of task offloading and computing, we introduce the Age-of-Information (AoI) [15] as a performance indicator in this article. AoI is a performance metric used to depict the freshness of information contained in the computing results, which is defined as the time elapsed since the most recent task was generated [16]. Different from communication delay and UAV flight time minimization, minimizing the AoI-related metrics can maximize the information freshness of the computing tasks, thereby ensuring the validity of the information for accurate decision-making.

In this article, given limited communication and computing resources as well as computing deadline requirements in the AIRS-assisted MEC networks, we aim to minimize the average AoI and energy consumption of the system. To this end, we tend to improve information freshness and curtail energy expenditure by optimizing the device association, scheduling for offloading, IRS phase shift, and allocations of communication and computing resources. To reflect the unpredictable offloading in the AIRS-assisted MEC network practically, we consider the random arrival of computing tasks when building the joint optimization problem. The main contributions of this article are summarized as follows.

1) We propose a novel AIRS-assisted offloading model under given available resources to address the challenges of offloading and computing with timeliness and reliability guarantees in MEC networks where the direct links between the devices and the HAP are poor.
2) To reveal the potential of AIRSs in improving timeliness and saving energy, we formulate a minimization problem of average AoI of computing results and system energy consumption by optimizing the association between devices and AIRSs, IRS phase shift design, scheduling of computing tasks for offloading, and resource allocations.
3) We develop an efficient problem-solving framework and propose a low-complexity algorithm to solve the formulated problem, which can provide insights for dealing with intractable mixed-integer and nonconvex problems. The algorithm performance and the relations between resources are analyzed to provide design guidelines for task offloading and resource allocations.

The remainder of this article is organized as follows. Related works are summarized in Section II. The system model and problem formulation are presented in Sections III and IV, respectively. The proposed solution is detailed in Section V. Extensive simulation results are presented and analyzed in Section VI. Finally, this article is concluded in Section VII.

II. Related Work

Recent years have witnessed a wide range of MEC-related research that is enabled by different key technologies and considered from different concerns. Next, we will review and present research on MEC-related work, with a focus on enabling technologies and optimization objectives.

To enhance the performance of MEC networks, many prominent efforts have been made by integrating MEC and other technologies, e.g., UAV [17], [18], device-to-device (D2D) [19], wireless power transfer (WPT) [20], and multiantenna nonorthogonal multiple access (NOMA) [21]. Chen et al. [17] concentrated on the problem of AoI-aware task offloading in a multiaccess edge computing system, wherein the ground MEC server and the UAVs jointly provide computing services. Li et al. [18] studied the UAV-assisted MEC intending to optimize offloading with minimum UAV energy consumption. Xie et al. [19] investigated the collaborative task offloading of devices in a D2D-enabled MEC system and addressed a task-flow constrained network-wide utility maximization problem. A framework integrating the WPT technology with MEC was developed in [20] to charge for multiple users and execute tasks offloaded from users. Under this framework, the authors designed a resource allocation scheme based on the time division multiple access (TDMA) to minimize the total energy consumption of the HAP. Wang et al. [21] exploited the NOMA technology to enable offloading for multiple users. They tried to minimize the weighted sum-energy consumption for all users with computation latency constraints by optimizing the resource allocations and the decoding order of the base station. Among these efforts, most studies [18], [19], [20], [21] focus on energy consumption, with only a few [17] focusing on information freshness.

Given the widespread application of the IRS in the field of communications, we delve into MEC studies integrating the IRS, recognizing that IRS-assisted MEC research is still in its fancy. Despite a wealth of IRS-related studies [8], [9], [10], [11], [13], [22], [23], [24], [25], these works emphasize communication performance improvement, offering limited insights into MEC applications. To the best of our knowledge, several studies have explored terrestrial IRS-assisted MEC [26], [27], [28], [29], [30] which have revealed the benefits of IRS assistance in edge computing, but the research on AIRS-assisted MEC has been scarce for the time being. Chu et al. [26] studied an IRS-assisted wireless-powered MEC

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

and caching network, wherein a network utility maximization problem was formulated and solved by designing the IRS phase shift and caching strategy. An edge heterogeneous network with the assistance of IRS was proposed in [27] to improve the network performance. The authors minimized the long-term energy consumption subject to the constraints imposed on Quality of Service (QoS) and available resources by optimizing the user association, computation offloading, resource allocation, and IRS phase shift design. Bai et al. [28] investigated the pertinent latency-minimization problems for single-device and multidevice scenarios with limited edge computing capability to explore the benefits of IRSs in MEC systems. The IRS-based backscatter communication is studied in [29] to realize computational task offloading of energy-constrained MEC network in a self-sustainable manner. The authors achieved a higher communication performance by solving the formulated sum of a computational bits maximization problem. Chen et al. [30] studied the achievable computation rate performance of IRS-aided wireless-powered MEC systems and discussed which multiple access scheme is superior for offloading by considering the impact of IRS. The benefits of IRS in reducing the offloading latency and energy consumption, and alleviating the backhaul burden are evident, even if these efforts focus on only one concern. In addition, the potential of AIRS assistance in MEC networks is still unknown.

Motivated by these efforts, we attempt to deploy multiple AIRSs in MEC networks with poor offloading environments to provide sustainable, reliable, timely, and energy-saving offloading and computing for resource-constrained devices. Given that the QoS of MEC systems usually require multiple concerns, we construct a problem of minimizing the average AoI and energy consumption by considering the information freshness and energy consumption. This article aims to fill the research gap in integrating AIRSs and MEC, reveal the potential of AIRSs for MEC performance improvement, and study the relations between communication and computing resources in AIRS-assisted MEC networks.

III. System Model

We consider an AIRS-assisted MEC network, as shown in Fig. 1, which consists of a single-antenna HAP[1] equipped with an edge server, K IoT devices, and M AIRSs. The devices scattered throughout the network monitor the physical environment, capturing information, such as the status of a wireless sensor network or traffic conditions in an intelligent transportation system. These devices generate computing tasks containing the monitored information. Due to insufficient computing capability and limited energy budgets of the devices, the generated computing tasks need to be offloaded to the HAP for processing. AIRSs are deployed to assist in offloading when the direct links between devices and the HAP are poor.

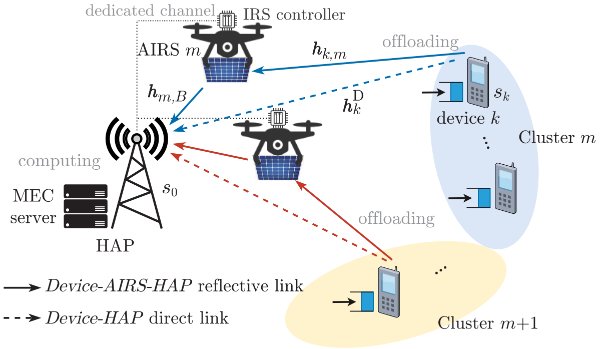


Fig. 1. System model of a MEC network assisted by multiple AIRSs.

The computing tasks are not separable due to the application requirements and data security considerations.

Each device is equipped with an antenna, while each AIRS is equipped with a uniform planar array (UPA) of F reflective elements. The phase shift of each AIRS is adjusted by the controller mounted on the UAV according to the results sent by the HAP. We utilize a 3-D Cartesian coordinate system, wherein the HAP is located at the origin $\boldsymbol{s}_0 = (0, 0, H)$, where H indicates the antenna height of the HAP. The position of the kth device is denoted by $\boldsymbol{s}_k = (x_k, y_k, 0)\ \ \forall k \in \mathcal{K} \triangleq \{1, \ldots, K\}$. The deployment positions of AIRSs are known[2] in advance. Given that the AIRS positions might be dynamic as it follows a designated route, the hovering position of the mth AIRS in time slot t is denoted by $\mathbf{q}_m[t] = (x_m[t], y_m[t], h)\ \ \forall m \in \mathcal{M} \triangleq \{1, \ldots, M\}$. Without loss of generality, all AIRSs fly at the same altitude.

A. Computing Task Generation

Within the allowable range of endurance for all UAVs, the time horizon of this system is divided into T equally spaced time slots, defined as $\mathcal{T} = \{1, \ldots, T\}$. Each time slot lasts for δ seconds, which indicates the deadline for processing each computing task. The generation process of the computing task for each device is modeled as a Bernoulli process [22]. Specifically, a computing task is generated at each device with a probability $\zeta_k \in (0, 1]$, i.e., $P(r_k[t] = 1) = \zeta_k$, where $r_k[t]$ is a binary indicator to characterize whether a task is generated at the beginning of the time slot t or not. The task generation processes of different devices are independent of each other. The information validity can be figured out by the HAP after the computing is finished at the end of one time slot.

B. Association and Scheduling Model

The offloading is realized in the nonoverlapping channel through frequency division multiplexing. The total available bandwidth of the system is divided into M equal sub-bands. Each sub-band is assigned for use by an AIRS, which can facilitate up-link channel estimation for obtaining channel

[1] To purely unveil the potential of the AIRSs, the HAP is equipped with one antenna. The solution obtained in this article is also applicable to the case of HAP equipped with multiple antennas. For instance, the receive beamformers conceived for the TDMA and NOMA schemes are maximum ratio combiner [20] and minimum mean square error-based arrangements [21], respectively.

[2] The deployment of AIRSs is planned in advance according to the geographical environment. The AIRS positions can be known through global position system (GPS) positioning or be estimated through aerial photography target detection [31] and share the information with the HAP.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

state information (CSI)[3] and avoid co-channel interference for ensuring offloading reliability and information security.

While the HAP is aware of the locations of devices scattered over a large Region of Interest (RoI), scheduling one device throughout the entire RoI within each time slot for offloading poses a challenge for individual AIRS with limited onboard battery capacity. To reduce the complexity of global scheduling for offloading and prolong the UAV endurance, we first solve the association between devices and AIRSs, so that each AIRS schedules a device from a limited number of devices associated with it within each time slot. Since the communication rate determines the reliability of offloading, the clustering method with the maximum signal-to-noise ratio (SNR) principle can achieve a good enough offloading rate and decrease the risk of flight collision for adjacent AIRSs. The classic K-means clustering method [33] is first applied to initially cluster all devices into M groups. The devices in cluster m, defined as $\mathcal{S}_m$, are associated with the mth AIRS, which is closest to the geometric center of the cluster. The number of devices in $\mathcal{S}_m$ is denoted by $K_m = |\mathcal{S}_m|$. As a result, each device has an AIRS assistance, i.e., $\sum_{m=1}^{M} K_m = K$.

Based on the initial association, we introduce a binary variable $x_{m,k}[t] = 1$ to indicate that the kth device is associated with the mth AIRS for offloading the computing task to the HAP, otherwise, $x_{m,k}[t] = 0$. To ensure a channel with sufficient quality for offloading, we impose the constraint that each AIRS serves at most one device within each time slot. The scheduling model for task offloading between devices in $\mathcal{S}_m$ and the AIRS is expressed by

$$x_{k,m}[t] \in \{0, 1\}, \sum_{k=1}^{K_m} x_{k,m}[t] \leq 1 \ \forall k \in \mathcal{S}_m, m \in \mathcal{M}, t \in \mathcal{T}. \quad (1)$$

C. Channel Model

Define the diagonal phase shift of the mth AIRS in time slot t as $\Theta_m[t] = \text{diag}\{e^{j\theta_{m,1}[t]}, \cdots, e^{j\theta_{m,F}[t]}\} \in \mathbb{C}^{F\times F}$, where $\theta_{m,f}[t]$ indicates the phase shift of the fth diagonal element, which needs to satisfy

$$\theta_{m,f}[t] \in [0, 2\pi) \ \ \forall m \in \mathcal{M}, t \in \mathcal{T}, f \in \mathcal{F} \quad (2)$$

where $\mathcal{F} = \{1, \ldots, F\}$. The reflective panels on UAVs are placed parallel to the ground. In more detail, the indices along the X and Y axes for each AIRS element are denoted by $0 \leq f_x \leq F_x$ and $0 \leq f_y \leq F_y$, respectively, where $F = F_x F_y$ and $f = F_y(f_x - 1) + f_y$.

Considering that a signal reflected by one AIRS toward the HAP typically tilts down to the ground, the signal reflections between AIRSs are neglected [8] due to the successive path loss. There are two types of communication links between each device and the HAP: 1) a direct *device-HAP* link and 2) a reflective *device-AIRS-HAP* link. We denote the channel gains between the kth device and the HAP, the kth device and the mth AIRS, the mth AIRS and the HAP by $\boldsymbol{h}_k^D[t] \in \mathbb{C}$, $\boldsymbol{h}_{k,m}[t] \in \mathbb{C}^{F\times 1}$, and $\boldsymbol{h}_{m,B}[t] \in \mathbb{C}^{F\times 1}$, respectively.

1) Device-HAP Link: The direct link $\boldsymbol{h}_k^D[t]$ is modeled as

$$\boldsymbol{h}_k^D[t] = \sqrt{\rho d_{k,B}^{-\alpha}}\bar{\boldsymbol{h}}_{k,B}[t] \ \ \forall k \in \mathcal{S}_m \quad (3)$$

where $\sqrt{\rho d_{k,B}^{-\alpha}}$ is the large-scale fading coefficient, α is the path loss exponent that usually has a value between 2 and 6, and ρ is the average channel gain for path loss at a reference distance of one meter. In (3), $d_{k,B} = \sqrt{\|\boldsymbol{s}_k - \boldsymbol{s}_0\|_2^2 + H^2}$ is the distance between the kth device and HAP. $\bar{\boldsymbol{h}}_{k,B}[t]$ is small-scale fading which is modeled as a Rayleigh fading channel with zero mean and unit variance [22].

2) Device-AIRS-HAP Link: Given that UAVs fly at a sufficiently high altitude where high-probability LoS links [34] can be established, the reflective links are modeled by the Rician fading channel with a dominant LoS [13], [35]. Thus, $\boldsymbol{h}_{k,m}[t]$ is modeled as

$$\boldsymbol{h}_{k,m}[t] = \sqrt{\rho d_{k,m}^{-\alpha}[t]}\sqrt{\frac{\kappa_1}{\kappa_1 + 1}}\bar{\boldsymbol{h}}_{k,m}\big(\phi_{k,m}^r[t], \eta_{k,m}^r[t]\big) \quad (4)$$

where $\sqrt{\rho d_{k,m}^{-\alpha}[t]}$ is the path loss coefficient, κ_1 is the Rician factor for $\boldsymbol{h}_{k,m}[t]$ link. $d_{k,m}[t] = \sqrt{\|\mathbf{q}_m[t] - s_k\|_2^2 + h^2}$ is the distance between the mth AIRS and the kth device. $\phi_{k,m}^r[t]$ and $\eta_{k,m}^r[t]$ are the azimuth angle and the elevation steering angle from the kth device to the mth AIRS, respectively.

Denote the carrier wavelength and space between the IRS elements by λ and d, respectively. The steering vector $\bar{\boldsymbol{h}}_{k,m}(\phi_{k,m}^r[t], \eta_{k,m}^r[t])$ is given in (5), shown at the bottom of the next page, where $\otimes$ indicates the Kronecker product. The superscripts $(\cdot)^H$ and $(\cdot)^{\text{T}}$ indicate the Hermitian transpose and the transpose, respectively. $u_{k,m}^r[t] = \sin(\phi_{k,m}^r[t])\cos(\eta_{k,m}^r[t])$ and $w_{k,m}^r[t] = \sin(\phi_{k,m}^r[t])\sin(\eta_{k,m}^r[t])$. Likewise, $\boldsymbol{h}_{m,B}[t]$ is modeled as

$$\boldsymbol{h}_{m,B}[t] = \sqrt{\rho d_{m,B}^{-\alpha}[t]}\sqrt{\frac{\kappa_2}{\kappa_2 + 1}}\bar{\boldsymbol{h}}_{m,B}\big(\phi_{m,B}^t[t], \eta_{m,B}^t[t]\big) \quad (6)$$

where $\sqrt{\rho d_{m,B}^{-\alpha}[t]}$ represents the path loss coefficient. κ_2 is the Rician factor for $\boldsymbol{h}_{m,B}[t]$ link, and $d_{m,B}[t] = \sqrt{\|\mathbf{q}_m[t] - s_0\|_2^2 + (h - H)^2}$ is the distance between the mth AIRS and the HAP in time slot t. $\phi_{m,B}^t[t]$ and $\eta_{m,B}^t[t]$ are the azimuth angle and the elevation steering angle from the mth AIRS to the HAP, respectively. The calculation of the steering vector $\bar{\boldsymbol{h}}_{m,B}(\phi_{m,B}^t[t], \eta_{m,B}^t[t])$ is similar to $\bar{\boldsymbol{h}}_{k,m}(\phi_{k,m}^r[t], \eta_{k,m}^r[t])$, which is omitted for brevity, where $u_{m,B}^t[t] = \sin(\phi_{m,B}^t[t])\cos(\eta_{m,B}^t[t])$ and $w_{m,B}^t[t] = \sin(\phi_{m,B}^t[t])\sin(\eta_{m,B}^t[t])$. Thus, the reflective channel gain between the kth device and the HAP via the mth AIRS in time slot t is given by

$$\boldsymbol{h}_k^R[t] = \boldsymbol{h}_{m,B}^H[t]\Theta_m[t]\boldsymbol{h}_{k,m}[t] \ \ \forall k \in \mathcal{S}_m, m \in \mathcal{M}, t \in \mathcal{T}. \quad (7)$$

D. Offloading Model

The offloading power of the kth device in time slot t, which is denoted by $p_{k,m}[t]$, is subject to

$$0 \leq p_{k,m}[t] \leq x_{k,m}[t]P_{\max} \ \forall k \in \mathcal{S}_m, t \in \mathcal{T} \quad (8)$$

[3] The system works in a legitimate network. The perfect CSI is assumed to be available based on the existing channel estimation techniques for IRS-assisted communications [26], [32]. The solution obtained in this article is deemed to be a best case performance bound in realistic scenarios.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

where $P_{\max}$ indicates the maximum radio frequency output power. The received SNR at the HAP is expressed by

$$\gamma_{k,m}[t] = \frac{p_{k,m}[t]\left|\boldsymbol{h}_k^D[t] + \boldsymbol{h}_k^R[t]\right|^2}{\Gamma\sigma^2} \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T} \tag{9}$$

where Γ indicates the SNR gap due to the practical modulation and coding scheme employed, and σ^2 is the variance of adding white Gaussian noise power at the receiver. The Doppler effect due to the UAV mobility is assumed to be well compensated by physical layer designs. Due to the positions of all facilities being static, the channel characteristics are assumed to be unchanged during the offloading time. The achievable offloading rate [36] of the kth device in time slot t is given by

$$R_{k,m}[t] = B\log_2\left(1 + \gamma_{k,m}[t]\right) \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T} \tag{10}$$

where B is the bandwidth of the preassigned frequency band.

We denote the computing task of the kth device by a tuple (D_k, C_k), where D_k represents the data size of the computing task and C_k represents the required computing resources for executing one-bit of input data. The task offloading time $\tau_k^{\text{off}}[t]$ within time slot t is given by

$$\tau_k^{\text{off}}[t] = \frac{D_k}{R_{k,m}[t]} \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T}. \tag{11}$$

The edge server at the HAP has multiple virtual queues that can perform parallel computations. The computing time for the offloaded task within time slot t is expressed as

$$\tau_k^{\text{edg}}[t] = \frac{D_k C_k}{f_m[t]} \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T} \tag{12}$$

where $f_m[t]$ is the computing resources allocated by the HAP to the virtual computing queue corresponding to the mth AIRS, which satisfies

$$\sum_{m=1}^{M} f_m[t] \le \bar{f} \quad \forall t \in \mathcal{T} \tag{13}$$

where $\bar{f}$ represents the total computing resource of the HAP. Additionally, the processing deadline requires that the offloading and computing for one device must be completed in one time slot to keep the information up-to-date, i.e.,

$$x_{k,m}[t]\left(\frac{D_k}{R_{k,m}[t]} + \frac{D_k C_k}{f_m[t]}\right) \le \delta \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T}. \tag{14}$$

Given the relatively small data size of the computing result, the feedback time for the results is negligible.

E. Energy Model

The circuit energy used to generate the computing task is ignored. The energy consumption of the kth device in time slot t used to offload the computing task is expressed by

$$e_k^{\text{off}}[t] = \frac{D_k p_{k,m}[t]}{R_{k,m}[t]} \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T}. \tag{15}$$

Based on the dynamic voltage and frequency scaling [5] technique, the computing energy consumed at the HAP is

$$e_k^{\text{cop}}[t] = \xi D_k C_k f_m^2[t] \quad \forall k \in \mathcal{S}_m, t \in \mathcal{T} \tag{16}$$

where ξ is the effective capacitance coefficient which is determined by the processor chip architecture.

The AIRSs hover at the deployment positions for propulsion energy savings and signal stability. According to the most commonly used energy consumption model [36], the power consumption of the AIRS during hovering is a constant, denoted by P_m^H, depending on the rotor disc area, aircraft weight, and air density, etc. The IRS-bearing energy consumption can be reflected by P_m^H. The energy consumption related to offloading for the mth AIRS is a constant c_m, including the energy consumption of regulating the IRS phase shift and circuit. Thus, the total energy consumed by AIRSs is

$$e^{\text{UAV}} = \sum_{t=1}^{T}\sum_{m=1}^{M}(\delta P_m^H + c_m) = \sum_{m=1}^{M} T(\delta P_m^H + c_m). \tag{17}$$

Consequently, during the system time permitted by the UAV endurance, the total energy consumption is expressed as

$$E_{\text{total}} = \sum_{t=1}^{T}\sum_{m=1}^{M}\sum_{k=1}^{K_m} x_{k,m}[t](e_k^{\text{off}}[t] + e_k^{\text{cop}}[t]) + e^{\text{UAV}}. \tag{18}$$

Since e^{UAV} is not determined by the optimization variables related to offloading and computing, thus it is omitted in the following optimization process.

F. AoI Evolution Model

AoI is determined by the time waiting for task offloading and the time slot imposed by the offloading and computing. Consider that there is no backlogged task to be processed for each device, which means that the newly arrived computing task will replace the task waiting for offloading. Recall that $r_k[t] = 1$ indicates that a computing task of the kth device is generated in time slot t. The waiting time of the computing task for the kth device at the beginning of time slot t, denoted by $z_k[t]$, evolves as follows:

$$z_k[t+1] = \begin{cases} 0, & \text{if } r_k[t+1] = 1 \\ z_k[t] + 1, & \text{otherwise} \end{cases} \tag{19}$$

$$\begin{aligned} \bar{\boldsymbol{h}}_{k,m}[t] = {} & \left(1, \ldots, e^{-j\frac{2\pi d}{\lambda} f_x \sin(\phi_{k,m}^r[t]) \cos(\eta_{k,m}^r[t])}, \ldots, e^{-j\frac{2\pi d}{\lambda}(F_x - 1)\sin(\phi_{k,m}^r[t])\cos(\eta_{k,m}^r[t])}\right) \\ & \otimes \left(1, \ldots, e^{-j\frac{2\pi d}{\lambda} f_y \sin(\phi_{k,m}^r[t]) \sin(\eta_{k,m}^r[t])}, \ldots, e^{-j\frac{2\pi d}{\lambda}(F_y - 1)\sin(\phi_{k,m}^r[t])\sin(\eta_{k,m}^r[t])}\right), \\ = {} & \left[1, \ldots, e^{-j\frac{2\pi d}{\lambda}(f_x u_{k,m}^r[t] + f_y w_{k,m}^r[t])}, \ldots, e^{-j\frac{2\pi d}{\lambda}\left((F_x - 1)u_{k,m}^r[t] + (F_y - 1)w_{k,m}^r[t]\right)}\right]^{\mathrm{T}}. \end{aligned} \tag{5}$$

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

for all $k \in \mathcal{S}_m, t \in \bar{\mathcal{T}}$. $\bar{\mathcal{T}}$ is the set of elements in $\mathcal{T}$ except T. A binary variable $a_k[t]$ is introduced to indicate whether there is an available packet in the kth device at the beginning of time slot t or not [22]. When the computing task of the kth device is offloaded and computed successfully in the previous time slot, meanwhile, there is no new arrival in the current slot, $a_k[t+1] = 0$. If there is a new arrival at the beginning of time slot $t+1$, $a_k[t+1] = 1$. For other cases, $a_k[t+1] = a_k[t]$. The evolution of the packet available status is expressed by

$$a_k[t+1] = \begin{cases} 0, & \text{if } r_k[t] = 1, x_{k,m}[t] = 1 \\ & \tau_k^{\text{off}}[t] + \tau_k^{\text{edg}}[t] \le \delta, r_k[t+1] = 0 \\ 1, & \text{if } r_k[t+1] = 1, \\ a_k[t], & \text{otherwise} \end{cases} \tag{20}$$

for all $k \in \mathcal{S}_m, t \in \bar{\mathcal{T}}$. Based on this, the indicator of the available packet at each time slot is expressed by

$$a_k[t+1] = r_k[t+1] + a_k[t](1 - x_{k,m}[t])(1 - r_k[t+1]) \tag{21}$$

for all $t \in \bar{\mathcal{T}}$. If a packet of the kth device is scheduled for offloading at the beginning of time slot t before a new packet arrives and is offloaded and computed successfully in time slot t, its AoI in the subsequent time slot will drop to $z_k[t] + 1$. Otherwise, AoI will increase gradually over time. Based on the initial association results, the AoI of the kth device in $\mathcal{S}_m$ is denoted by $A_{k,m}[t]$. The expression of AoI evolution process for the kth device during T time slots is given by

$$A_{k,m}[t+1] = \begin{cases} z_k[t] + 1, & \text{if } a_k[t] = 1, x_{k,m}[t] = 1 \\ \tau_k^{\text{off}}[t] + \tau_k^{\text{edg}}[t] \le \delta, & \\ A_{k,m}[t] + 1, & \text{otherwise} \end{cases} \tag{22}$$

for all $k \in \mathcal{S}_m, t \in \bar{\mathcal{T}}$. Under the premise that (14) is satisfied, the AoI of the device k is expressed by

$$\begin{aligned} A_{k,m}[t+1] = {} & x_{k,m}[t] a_k[t] z_k[t] + \\ & (1 - x_{k,m}[t]) a_k[t] A_{k,m}[t] + 1 \end{aligned} \tag{23}$$

for all $k \in \mathcal{S}_m, t \in \bar{\mathcal{T}}$. The average value of the sum of AoI for all devices over T time slots is given by

$$\text{AO} = \frac{1}{KT} \sum_{t=1}^{T} \sum_{m=1}^{M} \sum_{k=1}^{K_m} A_{k,m}[t]. \tag{24}$$

An example shown in Fig. 2 visualizes the AoI evolution process. In this example, we consider the device k with a computing task arrival probability of 0.2. The initial value of AoI is 1, i.e., $A_{k,m}[0] = 1$. The first computing task is generated at $t = 4$ when there is no other task in the device. The waiting time $z_k[4]$ of the current task is 0. z_k will gradually increase with time until the first task is scheduled at $t = 8$. The subsequent offloading succeeds, thus $z_k[8] = 0$ and the AoI value drops from a peak of 9 to 5. After that, a new computing task arrives at $t = 9$. AoI continues to grow until the second task is scheduled for offloading and computed in time slot $t = 14$, where the AoI value reaches a new peak, i.e., $A_{k,m}[14] = 10$. Then, $A_{k,m}$ drops to 6 at $t = 15$. Likewise, the AoI evolution regarding the third, fourth, and fifth tasks is similar to the above process.

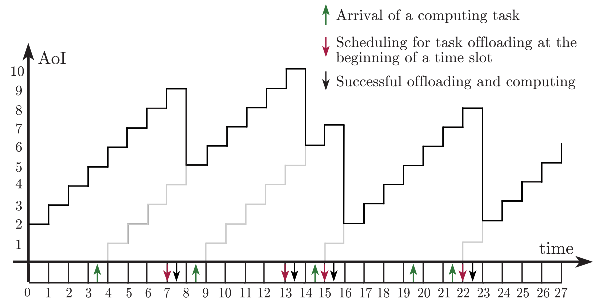


Fig. 2. Example of the AoI evolution over time for the kth IoT device.

IV. Problem Formulation

The optimization objective of this article is to improve the freshness of information and reduce the energy consumption of the system. Thereby, we utilize a simple additive weighting method to merge the average AoI and the total energy consumption into a single utility function for easier handling. The corresponding optimization problem is formulated as

$$\begin{aligned} \text{P}_1: \quad & \min_{\substack{\{\theta_{m,f}[t], x_{k,m}[t], \\ p_{k,m}[t], f_m[t]\}}} \; w_1 \, \text{AO} + w_2 E_{\text{total}} \\ \text{s.t.} \quad & (1), (2), (8), (13), (14) \end{aligned} \tag{25}$$

where w_1 and w_2 are the constant weights assigned to AO and E_{total}, respectively, which can be adjusted based on the system's preference for timeliness and energy efficiency. Problem P_1 is a mixed-integer nonconvex programming that cannot be directly solved by the existing convex optimization techniques, and it is intractable to obtain the optimal solution even with exhaustive searching.

The challenges of solving Problem P_1 mainly lie in two aspects. First, the optimization variable $x_{k,m}[t]$ is binary, which leads to Problem P_1 involving integer constraints. Second, the complex coupling relation between optimization variables makes Problem P_1 nonconvex. To solve these challenges, a series of transformations are applied to make Problem P_1 solvable, which are detailed in Section V.

V. Proposed Solution

To solve Problem P_1 efficiently, we propose an efficient problem-solving framework, illustrated in Fig. 3. First, we determine the optimal IRS phase shift design through numerical analysis for a specified AIRS deployment. Second, based on the initial K-means clustering results and the closed-form IRS phase shift design, we optimize the association between the AIRSs and IoT devices to improve the fairness of device offloading and balance the network load. Third, based on the results obtained in the previous two steps, we decompose Problem P_1 into two subproblems, i.e., task scheduling offloading and resource allocations, and solve them alternatively. Both optimizing the task offloading order and the resource allocations can be decomposed into M parallel second-level subproblems, each of which is solved sequentially by time slot. As long as there is no change in the environment, e,g., the number and locations of IoT devices or AIRSs, the aforementioned steps 1 and 2 do not need to

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

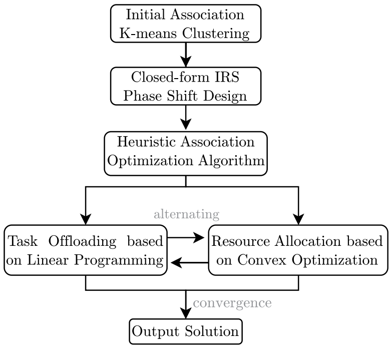


Fig. 3. Block diagram of the proposed solution for Problem P_1.

be conducted repeatedly. The proposed framework can greatly reduce the computational complexity of algorithm design, especially when offloading operates in large-scale scenarios.

A. *Optimal Phase Shift Design*

To ensure reliable and timely task offloading within each time slot, AIRSs should maximize the channel quality by maximizing the channel gains between each device and the HAP. The total channel gains maximization can be realized by maximizing the channel gains of reflective links via adjusting the IRS phase shift, which is achieved by optimizing

$$\begin{aligned}\mathrm{P}_2 \ : \ &\min_{\Theta_m[t]} \ |\boldsymbol{h}_k^R[t]|^2\\ &\text{s.t.} \ \ \theta_{m,f}[t] \in [0, 2\pi) \ \ \forall k \in \mathcal{S}_m, m \in \mathcal{M}, f \in \mathcal{F}, t \in \mathcal{T}.\end{aligned} \tag{26}$$

Given $\{\mathbf{q}_m[t]\}$, we aim to find the feasible IRS phase shift $\{\theta_{m,f}[t]\}$ that can maximize the reflective channel gain between the kth device and the HAP via the mth AIRS, which is rewritten as

$$\begin{aligned}\boldsymbol{h}_k^R[t] &= \boldsymbol{h}_{m,B}^H[t]\Theta_m[t]\boldsymbol{h}_{k,m}[t]\\ &= \frac{\rho\beta\sum_{f=1}^{F} e^{j\left(\theta_{m,f}[t]+\varphi_{m,B,f}[t]-\psi_{k,m,f}[t]\right)}}{\sqrt{d_{k,m}^{\alpha}[t]d_{m,B}^{\alpha}[t]}}\end{aligned} \tag{27}$$

where $\beta = \sqrt{(\kappa_1\kappa_2/[(\kappa_1+1)(\kappa_2+1)])}$, $\psi_{k,m,f}[t] = (-2\pi d/\lambda)(f_x u_{k,m}^r[t] + f_y w_{k,m}^r[t])$, and $\varphi_{m,B,f}[t] = (-2\pi d/\lambda)(f_x u_{m,B}^t[t] + f_y w_{m,B}^t[t])$. The optimal solution of Problem P_2 is obtained according to Lemma 1.

Lemma 1: For given $\{\mathbf{q}_m[t], x_{k,m}[t], p_{k,m}[t]\}$, the optimal solution of Problem P_2 is achieved when the IRS phase shift satisfies $\theta_{m,f}[t] = \psi_{k,m,f}[t] - \varphi_{m,B,f}[t] \ \ \forall k, m, f, t$.

Proof: Applying the triangle inequality to the UPA gain of AIRS m, we obtain the following inequality:

$$\begin{aligned}&\left|\sum_{f=1}^{F} e^{j\left(\theta_{m,f}[t]+\varphi_{m,B,f}[t]-\psi_{k,m,f}[t]\right)}\right|\\ &\quad\le \left|e^{j\left(\theta_{m,1}[t]+\varphi_{m,B,1}[t]-\psi_{k,m,1}[t]\right)}\right|\\ &\qquad+\cdots+|e^{j\left(\theta_{m,f}[t]+\varphi_{m,B,f}[t]-\psi_{k,m,f}[t]\right)}|\\ &\qquad+\cdots+|e^{j\left(\theta_{m,F}[t]+\varphi_{m,B,F}[t]-\psi_{k,m,F}[t]\right)}| = F\end{aligned} \tag{28}$$

where the equality holds with

$$\begin{aligned}\theta_{m,f}[t] &= \psi_{k,m,f}[t] - \varphi_{m,B,f}[t]\\ &= \frac{2\pi d}{\lambda}(f_x u_{m,B}^t[t] + f_y w_{m,B}^t[t] - f_x u_{k,m}^r[t] - f_y w_{k,m}^r[t])\end{aligned} \tag{29}$$

for all $k \in \mathcal{S}_m, m \in \mathcal{M}, f \in \mathcal{F}, t \in \mathcal{T}$. ■

Based on Lemma 1, a closed-form solution of the IRS phase shift for any deployment positions of AIRS can be obtained. Considering the operating cost of the system, we discuss the achievability of optimal phase shift in Remark 1.

Remark 1: In general, the IRS contains many reflective elements, which makes continuous adjustment of phase shifts costly. As such, it is more cost-effective to adjust discrete phase shifts in IRS compared to the design of continuous phase shifts. The IRS phase is generally divided into $(2\pi/2^b)$ discrete values [37], where b is the number of bits used to characterize the phase. When the continuous phase shift is discrete with sufficiently high precision, its performance can attain the optimal phase shift. The IRS phase shift obtained in this article provides a best case bound that can be approached through practical discrete phase shift adjustment.

B. *Association Optimization*

Although K-means clustering guarantees the offloading rate, unbalanced clustering will affect the fairness of device offloading and waiting time for offloading, thereby decreasing resource utilization and information freshness. Therefore, we develop a load balance approach based on the initial clustering results to cope with uneven computing demand distribution [38] caused by the difference in the load of each AIRS. The load of AIRS indicates the number of devices associated with it. Inspired by the coalition formation game [39], we design a transfer rule within the acceptable rate loss to transfer devices deliberately from heavy-loaded clusters to light-loaded clusters for load balance. The transferring rule is given by

$$R_{k,j} > R_{k,i} - R_\Delta \ \ \forall k \in C_i, i, j \in \mathcal{M}, i \neq j \tag{30}$$

where R_Δ is the acceptable loss of the offloading rate. Coalition m, denoted by C_m, indicates the set of devices associated with the mth AIRS. The utility of device k in coalition C_m is given by $U_{k,m} = \log(R_{k,m})$ [40], and the utility of C_m is

$$U_m(C_m) = \sum_{k\in C_m} \log(R_{k,m}) \ \ \forall m \in \mathcal{M}. \tag{31}$$

Then, the coalition game can be expressed by $\{\mathcal{C}, \mathcal{U}, \mathcal{M}, \mathcal{K}\}$, where $\mathcal{C}$ indicates the coalition set and $\mathcal{U}$ represents the set of utilities U for all devices. $C_{\max}$ and $C_{\min}$ represent the coalitions with the largest and smallest values of coalition utility, respectively. We introduce U_Δ to quantify whether the clustering is balanced. The association needs to be adjusted when the maximum difference between the utility of coalitions is greater than U_Δ, i.e.,

$$U(C_{\max}) - U(C_{\min}) \ge U_\Delta. \tag{32}$$

The details of association optimization are presented in Algorithm 1, where Φ indicates the empty set. The devices

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 Heuristic Association Optimization Algorithm

```
Input: tolerance of utility gap $U_\Delta$ and rate reduction $R_\Delta$.
Initialize coalitions $C_m = \mathcal{S}_m, \forall m$;
Calculating device utilities $\{U_{k,m}, \forall k \in C_m\}$ for all $m$;
Calculating coalition utilities $\{U_m(C_m), \forall m\}$;
Select the heavy-loaded coalition $C_{\max}$ with $\max\{U_m, \forall m\}$
    and light-loaded coalition $C_{\min}$ with $\min\{U_m, \forall m\}$;
Set set $\mathcal{W} = \Phi$ to store transferred devices;
while $U(C_{\max}) > U(C_{\min}) + U_\Delta$ do
    Set $j$ to the index of coalition $C_{\min}$;
    Set $i$ to the index of coalition $C_i, \forall i \neq j$;
    Update the set $\mathcal{T}_r$ by removing transferred devices in
        set $\mathcal{W}$ to avoid duplicate transfers;
    Select the device $k$ with the maximum transfer utility
        $\max\{U_{k,j}, \forall k \in C_i\}$ from the set $\mathcal{T}_r$;
    Update the set $\mathcal{T}_r$ by removing the device $k$;
    Update the set W by adding the device $k$;
    Update $C_i$ and $C_j$ by transferring device $k$ from
        $C_i$ to $C_j$;
    Calculate coalition utilities $U(C_i)$ and $U(C_j)$;
    Select the coalitions $C_{\max}$ with $\max\{U(C_m), \forall m\}$ and
        $C_{\min}$ with $\min U(C_m), \forall m$;
end while
Output: $\mathcal{S}_m = C_m, \forall m \in \mathcal{M}$.
```

TABLE I
Achieved Network Utility Under Different Ways of Association

Method	AIRS 1	AIRS 2	AIRS 3
K-means Clustering	10.56	24.51	17.63
Association Adjustment	17.60	17.48	17.60

satisfying (30) are concentrated in the set of transferable devices $\mathcal{T}_r$. In each iteration, we select and transfer the device k with the maximum transfer utility, i.e., $\max\{U_{k,j} \forall k\}$, from the set $\mathcal{T}_r$ to the minimal coalition $C_{\min}$. Then, coalitions are updated after transferring. C_i' and C_j' represent the coalitions after transferring the device k. An example of the comparison between the initial clustering and final association is given in Fig. 4, and the corresponding values of the network utility are given in Table I. The AIRS deployment position is the geometric center of its corresponding cluster, i.e., $\mathbf{q}_m[t] = \boldsymbol{s}_m \;\; \forall t$, where $\boldsymbol{s}_m$ is the geometric center of the cluster m. It is observed in Fig. 4 that the AIRS load is well balanced after the association optimization while the total network utility is almost unchanged. The convergence of Algorithm 1 has been analyzed in Theorem 1. The proposed algorithm can also be extended to situations where the movement of the quasi-static device has been known, which is explained in Remark 2. This facilitates the adaptation of dynamic networks.

Theorem 1: The coalition game built in this article can reach a stable coalition structure.

Proof: The number of game participants, i.e., K, are finite in this coalition game model, which indicates that the possible transfer options are limited. Based on the settled transfer rule, each transfer can guarantee that the maximum utility gap is reduced. Thus, this coalition game can converge to a coalition equilibrium structure [41]. ■

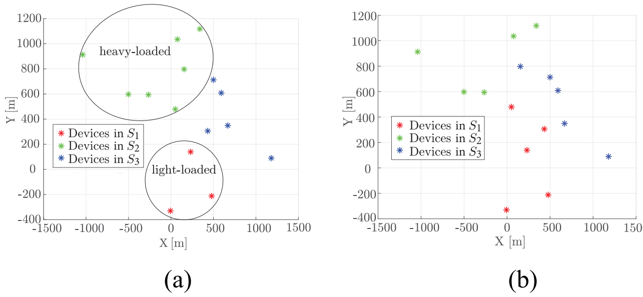


Fig. 4. Comparison between the initial clustering and the association adjustment results. S_k is the set of devices associated with the kth AIRS. (a) K-means clustering results. (b) Association adjustment.

Remark 2: Consider that the locations of the devices may change, the UAVs can calculate the device utility $U_{k,m}[t]$ at regular intervals and readjust the association with devices according to Algorithm 1. The time scale of algorithm optimization is determined according to the dynamic change characteristics of the network environment [42].

C. Optimization of Scheduling and Resource Allocation

Given the optimal IRS phase shift design and the association results, the original problem P_1 is simplified as

$$\begin{aligned}\text{P}_3 : \min_{\substack{\{x_{k,m}[t],\\ p_{k,m}[t], f_m[t]\}}} & \sum_{t=1}^{T}\sum_{m=1}^{M}\sum_{k=1}^{K_m}\Bigg(\frac{w_1 A_{k,m}[t]}{KT} + \\ & w_2 x_{k,m}[t]\Big(\frac{D_k p_{k,m}[t]}{R_{k,m}[t]} + \xi D_k C_k f_m^2[t]\Big)\Bigg) \qquad (33)\\ \text{s.t.}\ & (1),\ (8),\ (13),\ (14).\end{aligned}$$

Problem P_3 is mixed-integer and nonconvex due to the existence of binary variables and nonconvex constraint (14) caused by variable coupling. Fortunately, Problem P_3 can be decomposed into M parallel subproblems since there are no coupling constraints between AIRSs. Furthermore, considering the long system runtime, optimizing each parallel subproblem would entail a large number of variables. In sight of this issue, by analyzing the relation between AoI values in adjacent time slots, each subproblem can be solved sequentially by time slot to further reduce the computational complexity.

The objective for minimizing the average value of the sum of AoI among T time slots can be achieved by maximizing the AoI reduction within each time slot [23], [25]. Specifically, the device with the maximum AoI value should be scheduled at each time slot within each cluster to minimize the sum of the AoI among all devices. For the optimization of the mth subproblem within the time slot t, if a device is scheduled and its computing task is offloaded and computed before the deadline, its AoI reduction is $\sum_{k=1}^{K_m}(A_{k,m}[t] - z_k[t] - 1)x_{k,m}[t]$. Therefore, the mth parallel subproblem is further decomposed into T subproblems by the time slot, and the optimization problem within the time slot t is

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

Algorithm 2 Task Offloading Algorithm for Problem $\mathrm{P}_{4.1}$

```
Input: $w_1 = 1, i = 0, \epsilon > 0$.
Binary variable relaxation $x_{k,m}[t] \in [0, 1], \forall k \in \mathcal{S}_m$;
Initialize the feasible $\bar{f}_m[t]$, $\{\bar{p}_{k,m}[t], \forall k \in \mathcal{S}_m\}$;
repeat
    Set $i = i + 1$;
    Obtain $\{x^{(i)}_{k,m}[t], \forall k \in \mathcal{S}_m\}$ and the objective value $A^{(i)}_d$ by solving Problem $\mathrm{P}_{4.1}$;
until $\frac{|A^{(i)}_d - A^{(i-1)}_d|}{A^{(i-1)}_d} \leq \epsilon$;
Output: $A_d$ and $\{x_{k,m}[t], \forall k \in \mathcal{S}_m\}$.
```

Algorithm 3 Resource Allocation Algorithm for Problem $\mathrm{P}_{4.2}$

```
Input: $w_2 = 1, i = 0, \epsilon > 0$.
Initialize the feasible $\{\bar{x}_{k,m}[t], \forall k \in \mathcal{S}_m\}$;
repeat
    Set $i = i + 1$;
    Obtain $\{t^{off(i)}_{k,m}[t], e^{(i)}_{k,m}[t], f^{(i)}_m[t], \forall k\}$ and the objective value $E^{(i)}_{\text{total}}$ by solving Problem $\mathrm{P}_{4.2a}$;
    Calculate $p^{(i)}_{k,m}[t] = \frac{e^{(i)}_{k,m}[t]}{t^{off(i)}_{k,m}[t]}, \forall k \in \mathcal{S}_m$;
until $\frac{|E^{(i)}_{\text{total}} - E^{(i-1)}_{\text{total}}|}{E^{(i-1)}_{\text{total}}} \leq \epsilon$;
Output: $E_{\text{total}}$, $\{f_m[t], \forall m \in \mathcal{M}\}$, and $\{p_{k,m}[t], \forall k \in \mathcal{S}_m\}$.
```

written as

$$\mathrm{P}_4 : \min_{\substack{\{x_{k,m}[t],\\ p_{k,m}[t], f_m[t]\}}} \sum_{k=1}^{K_m} x_{k,m}[t]\Bigg(w_2\Big(\frac{D_k p_{k,m}[t]}{R_{k,m}[t]} + \xi D_k C_k f^2_m[t]\Big) - \frac{w_1(A_{k,m}[t] - z_k[t] - 1)}{K_m}\Bigg) \quad (34)$$
$$\text{s. t.} \quad (1),\ (8),\ (13),\ (14).$$

Problem P_4 is also mixed-integer and nonconvex, and it cannot be solved by a general-purpose solver through interior point methods. To make Problem P_4 solvable, we turn to alternately optimizing task offloading and resource allocations based on the block coordinate descent (BCD) technique [43].

1) Offloading Optimization: For given feasible resource allocation, i.e., $\{\bar{p}_{k,m}[t], \bar{f}_m[t]\}$, the difficulty of solving Problem P_4 lies in the existence of binary variables. To address the difficulty, we relax $x_{k,m}[t]$ into a continuous variable, i.e.,

$$x_{k,m}[t] \in [0, 1], \quad \sum_{k=1}^{K_m} x_{k,m}[t] \leq 1 \quad \forall k, m, t. \quad (35)$$

Consequently, Problem P_4 is transformed into

$$\mathrm{P}_{4.1} : \max_{x_{k,m}[t]} \frac{w_1}{K_m} \sum_{k=1}^{K_m} x_{k,m}[t](A_{k,m}[t] - z_k[t] - 1)$$
$$\text{s. t.} \quad (14),\ (35). \quad (36)$$

The optimization of device offloading is achieved by solving Problem $\mathrm{P}_{4.1}$, which is linear programming and can be solved by the existing convex solvers. The details for solving Problem $\mathrm{P}_{4.1}$ are outlined in Algorithm 2, where $A^{(i)}_d$ indicates the objective value in iteration i. Subsequently, $x_{k,m}[t]$ is recovered to binary according to the maximum principle in [44].

2) Resource Allocations: Based on the obtained offloading decision $\{x_{k,m}[t]\}$, resources are optimized by solving

$$\mathrm{P}_{4.2} : \min_{\substack{\{p_{k,m}[t],\\ f_m[t]\}}} \sum_{k=1}^{K_m} w_2 \bar{x}_{k,m}[t]\Big(\frac{D_k p_{k,m}[t]}{R_{k,m}[t]} + \xi D_k C_k f^2_m[t]\Big)$$
$$\text{s. t.} \quad (8),\ (13),\ (14). \quad (37)$$

Problem $\mathrm{P}_{4.2}$ is nonconvex due to the variable coupling. To solve the difficulty, we commence by handling the nonconvex constraint (14). We introduce a slack variable $t^{\text{off}}_{k,m}[t]$, where

$$t^{\text{off}}_{k,m}[t] \geq \frac{D_k}{R_{k,m}[t]} \quad \forall k \in \mathcal{S}_m. \quad (38)$$

Then, (14) is converted into a convex constraint by taking $t^{\text{off}}_{k,m}[t]$ into it. To avoid introducing additional variables to increase the computational complexity, we introduce the auxiliary variable $e_{k,m}[t] = t^{\text{off}}_{k,m}[t] p_{k,m}[t]$, which can substitute variable $p_{k,m}[t]$ equivalently in Problem $\mathrm{P}_{4.2}$.

By taking $e_{k,m}[t]$ into the introduced (38), we convert (38) into a equivalent but more tractable form, i.e.,

$$t^{\text{off}}_{k,m}[t] B \log_2\Big(1 + \frac{e_{k,m}[t] |H_k[t]|^2}{t^{\text{off}}_{k,m}[t]\sigma^2}\Big) \geq D_k \quad (39)$$

where $H_k[t] = \boldsymbol{h}^D_k[t] + \boldsymbol{h}^R_k[t] \quad \forall k \in \mathcal{S}_m$. Constraint (39) is a convex constraint since the function $f(x, y) = y \log(1 + [x/y]), y > 0$ is a perspective function of $\log(1 + x)$ which is concave.

Problem $\mathrm{P}_{4.2}$ is solved by solving the equivalent Problem $\mathrm{P}_{4.2a}$, which is formulated as

$$\mathrm{P}_{4.2a} : \min_{\substack{\{t^{\text{off}}_{k,m}[t],\\ e_{k,m}[t], f_m[t]\}}} \sum_{k=1}^{K_m} w_2 \bar{x}_{k,m}[t]\Big(e_{k,m}[t] + \xi D_k C_k f^2_m[t]\Big) \quad (40a)$$
$$\text{s. t.} \quad (8),\ (13),\ (39)$$
$$\bar{x}_{k,m}[t]\Big(t^{\text{off}}_{k,m}[t] + \frac{D_k C_k}{f_m[t]}\Big) \leq \delta. \quad (40b)$$

Problem $\mathrm{P}_{4.2}$ is convex and can be solved directly by the off-the-shelf convex solvers, e.g., CVX [45]. The details of solving Problem $\mathrm{P}_{4.2}$ is stated in Algorithm 3. The equivalence proof is presented in Theorem 2.

Theorem 2: Problem $\mathrm{P}_{4.2}$ is equivalent to Problem $\mathrm{P}_{4.2a}$.

Proof: Please see Appendix A. ■

To sum up, Problem P_3 is decomposed into M parallel subproblems, each of which is solved sequentially by time slot. This enables parallel computation over multicore CPUs and reduces the complexity of optimizing subproblems, thus the algorithm complexity and run-time are reduced.

D. Overall Algorithm

The above analysis focuses on developing efficient algorithms for task offloading and computing, the overall algorithm for solving Problem P_1 is summarized in Algorithm 4.

1) Algorithm Convergence: The convergence of Algorithm 1 for association optimization is guaranteed by Theorem 1. Simulation results prove that both Algorithms 2

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

Algorithm 4 Overall Algorithm for Solving Problem P_1

```
Input: weights $\{w_1, w_2\}$, iteration index $i = 0$, precision $\epsilon > 0$.
 1: Initialize the association $\{x_{k,m}, \forall k, m\}$ between IoT devices and AIRSs based on K-means clustering method;
 2: Calculate the optimal IRS phase shift $\{\theta_{m,f}[t], \forall m, f, t\}$ according to Lemma 1;
 3: Adjust the association $\{x_{k,m}, \forall k, m\}$ and update $\{\mathcal{S}_m, \forall m\}$ according to Algorithm 1;
 4: Initialize the feasible resource $\{p^{(0)}_{k,m}[t], f^{(0)}_m[t], \forall k, m, t\}$;
 5: Set the initial objective value $\mathrm{Obj}^{(0)} = \mathit{Inf}$;
 6: for $m = 1, \cdots, M$ do
 7:     for $t = 1, \cdots, T$ do
 8:         repeat
 9:             Set $i = i + 1$;
10:             Given $\bar{p}_{k,m}[t], \bar{f}_m[t]$, update $\{x^{(i)}_{k,m}[t], \forall k\}$ and $\mathrm{AO}^{(i)}$ by solving Problem $\mathrm{P}_{4.1}$ via Algorithm 2;
11:             Update $\bar{x}_{k,m}[t] = x^{(i)}_{k,m}[t], \forall k$;
12:             Given $\{\bar{x}_{k,m}[t], \forall k\}$, update $\{t^{\mathit{off}(i)}_{k,m}[t], e^{(i)}_{k,m}[t], f^{(i)}_m[t], \forall k\}$ and $E^{(i)}_{\mathrm{total}}$ by solving Problem $\mathrm{P}_{4.2a}$ via Algorithm 3;
13:             Calculate $p^{(i)}_{k,m}[t] = \frac{e^{(i)}_{k,m}[t]}{t^{\mathit{off}(i)}_{k,m}[t]}, \forall k$;
14:             Update $\bar{p}_{k,m}[t] = p^{(i)}_{k,m}[t], \bar{f}_m[t] = f^{(i)}_m[t], \forall k$;
15:             Update $\mathrm{Obj}^{(i)} = w_1\, \mathrm{AO}^{(i)} + w_2 E^{(i)}_{\mathrm{total}}$;
16:         until $\frac{|\,\mathrm{Obj}^{(i)} - \mathrm{Obj}^{(i-1)}\,|}{\mathrm{Obj}^{(i-1)}} \le \epsilon$
17:         Binary variable recovery $x_{k,m}[t] \in \{0, 1\}, \forall k \in \mathcal{S}_m$;
18:         Update $A_{k,m}[t]$ and $\{p_{k,m}[t], f_m[t]\}$ for all $k, m$ based on Eq. (23) and Algorithm 3, respectively;
19:     end for
20: end for
Output: AO, $E_{\mathrm{total}}$, and $\{x_{k,m}[t], p_{k,m}[t], f_m[t], t^{\mathit{off}}_{k,m}[t], \forall k, m, t\}$.
```

TABLE II
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
M	3	K	15
F	64	T	20
δ	0.5 s	κ_1, κ_2	10 dB
α	2.3	λ	0.1304
ζ	0.5	d	$\lambda/2$
h	100 m	H	20 m
B	10 KHz	$P_{\max}$	20 dBm
ρ	10 dBm	σ^2	-110 dBm
$\bar{f}$	10^8 cycles/s	ξ	10^{-26}
D_k	10 Kbits	C_k	10^3 cycles/bit

and 3 can converge within a few iterations under a given precision. The overall algorithm convergence is shown in Theorem 3 according to the execution process of Algorithm 4.

Theorem 3: The proposed overall Algorithm 4 is convergent.

Proof: Please refer to Appendix A. ■

2) Complexity Analysis: According to the proposed problem-solving framework illustrated in Fig. 3, the computational complexity of Algorithm 4 is primarily constructed by that of Algorithms 1–3. For Algorithm 1, within each iteration of the while-loop, the computational complexity is mainly determined by the calculation of device utility and coalition utility, the selection of coalition utility and the transfer devices, along with the update of coalition and device set [43]. The complexities of these operations are KM, $K^2 + M^2$, and K, respectively. Therefore, Algorithm 1 has a complexity of $\mathcal{O}(I_1(KM + K^2 + M^2))$, denoted by $\mathcal{O}_1$, where I_1 is the number of while-loop iterations. For Algorithms 2 and 3, the computational complexity is polynomial, which is determined by the number of iterations and the optimization variables for solving convex subproblems in each iteration. The computational complexity of Algorithm 2 in each iteration is $\mathcal{O}(K^3)$ since there are at most K variables optimized within each iteration. Thus, the total computational complexity is $\mathcal{O}(I_2K^3)$, denoted by $\mathcal{O}_2$, where I_2 indicates the number of iterations for achieving convergence. Likewise, the computational complexity of Algorithm 3 is $\mathcal{O}(I_3(K + M)^3)$, denoted by $\mathcal{O}_3$. Consequently, the total computational complexity for solving Problem P_1 is $\mathcal{O}_1 + \mathrm{MTI}_o(\mathcal{O}_2 + \mathcal{O}_3)$, where I_0 is the number of iterations required for convergence in Step 16 of Algorithm 4.

Remark 3: The proposed problem-solving framework in this article significantly reduces computational complexity and algorithm runtime. Without the proposed framework, solving Problem P_1 by converting it into an approximate convex problem has a computational complexity of at least $\mathcal{O}(I(\mathrm{MTF} + 2\mathrm{KMT} + \mathrm{MT})^3)$, where I indicates the iteration number for achieving convergence.

VI. Simulation Results

In this section, we present simulation results to evaluate the performance of the proposed solution. We consider a rectangular area with 2 km × 3 km, wherein 3 AIRSs assist 15 randomly distributed IoT devices in offloading tasks to the HAP at the position $(0, 0, 20)$ as a simulation example. We consider the general case where the system has no particular preference for information freshness and energy consumption. Unless stated otherwise, parameter settings are shown in Table II. Additionally, we verify the performance of the proposed solution by comparing it with several benchmarks, which are explained as follows.

1) *Random Offloading:* Each AIRS adopts a random offloading scheme to serve its associated devices. The association and resource allocation are optimized by referring to Algorithms 1 and 3, respectively.
2) *R-Robin Offloading:* Each AIRS adopts a Round Robin offloading scheme to serve its associated devices to ensure the fairness of offloading. The association and resource allocation are optimized by referring to Algorithms 1 and 3, respectively.
3) *Without AIRSs:* We consider the direct task offloading without the AIRSs assistance. The association optimization, offloading design, and resource allocation are optimized by referring to Algorithm 4.
4) *Fixed IRS-Phase:* We consider the fixed AIRS phase shift when assisting any device in task offloading. The association adjustment, offloading design, and resource allocation are optimized via Algorithm 4.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

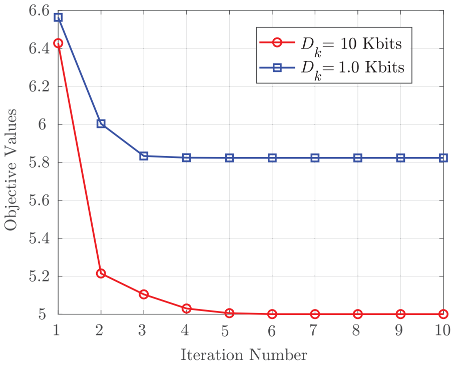


Fig. 5. Convergence behaviors of the algorithm for Problem P_1.

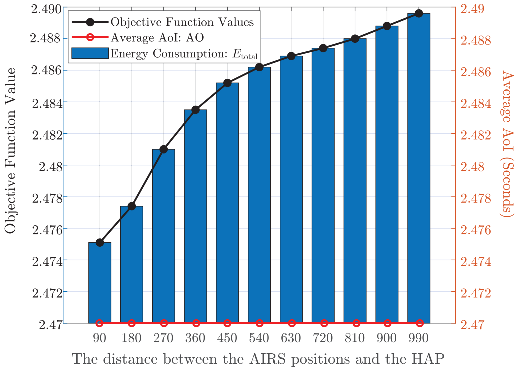


Fig. 6. Impact of the AIRS deployment on the objective.

5) *Fixed Res-Allocation:* We consider the fixed resource allocation, including the device power control and the HAP computing resource allocation. The association and offloading design are optimized by referring to Algorithms 1 and 2, respectively.

A. Algorithm Convergence Behavior

The convergence behavior of the proposed algorithm is presented in Fig. 5. It is observed that the objective function value gradually decreases as the number of iterations increases until it tends to a stable value. Even for the test under different parameter settings, e.g., D_k and F, the proposed algorithm can achieve rapid convergence within a few iterations.

B. Impact of AIRS Deployment

In this section, we reveal the impact of AIRS deployment on system performance. We take the devices in Cluster 2 as an example to analyze the impact of AIRS deployment on the AoI and energy consumption in Fig. 6. The AIRS is deployed on a straight line between the HAP and the geometric center of the cluster, where the X-axis indicates the horizontal distance between the AIRS and the HAP. It is observed in Fig. 6 that the objective function value increases with the horizontal distance between the AIRS and the HAP, but the increment is limited. The height of the blue rectangle indicates the total energy consumption of the system. We observe that the AoI value remains the same while the energy consumption increases when the deployment position changes. This is because the AIRS deployment affects the offloading time by the channel quality, thus affecting the energy consumption. The changes in offloading time are not enough to change the proportion of offloading time and computing time, and it does not affect the task offloading, so there is no impact on AoI.

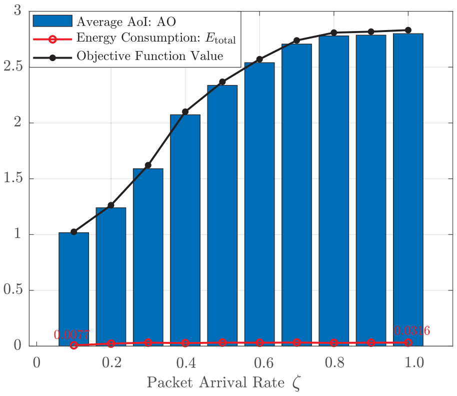


Fig. 7. Impact of the packet arrival rate ζ on the objective.

Furthermore, the increase in energy consumption due to the changes in the AIRS deployment position is not significant since the channel quality in this system is dominated by the UPA gains provided by the AIRS, in which case the impact of position on energy consumption is minimal. However, in scenarios that are very sensitive to energy consumption, it is best to deploy AIRS near the HAP for energy saving, which is consistent with the principle of terrestrial IRS deployment.

C. Impact of Task Arrival Rate and Task Size

We present and analyze the impact of the arrival rate ζ and data size D_k of the computing task on the objective function in this section. The impact of ζ on the information freshness and energy consumption is shown in Fig. 7. With the increase of arrival rate ζ, the average AoI AO, total energy consumption E_{total}, and the objective function increase significantly and eventually reach stable values. A high-arrival rate ζ means that there are more computing tasks to be scheduled, which leads to an increase in the number of offloaded tasks and inevitably increases the AoI and energy consumption.

The impact of D_k on the AoI and energy consumption is presented in Fig. 8. It is observed that D_k has little impact on the AoI since it basically does not affect the task offloading when (14) is satisfied within a time slot. However, the increase in task volume will affect the time of offloading and computing, thereby increasing energy consumption. The growth trend of the objective function and energy consumption is thus almost the same. Since the task size from different applications is different, it suggests weighing the volume of computing tasks when providing the communication and computing capabilities to save energy.

D. AoI Evolution

In this section, we take three devices in Cluster 3 as an example to show the evolution of AoI over time, which is illustrated in Fig. 9. Although the task arrival rate ζ for the three devices is the same, the AoI evolution for the three

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

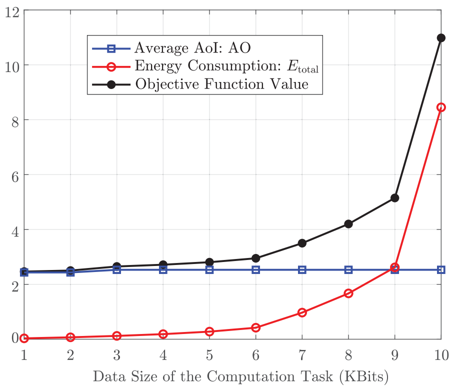


Fig. 8. Impact of the computing task size D_k on the objective.

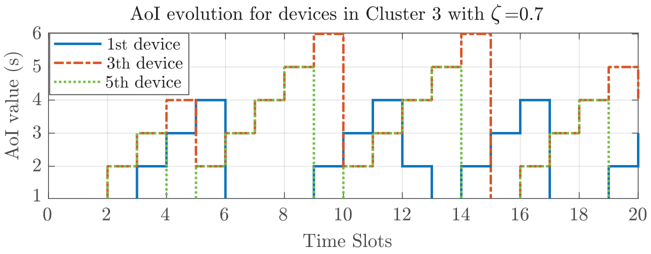


Fig. 9. Example of the AoI evolution over time.

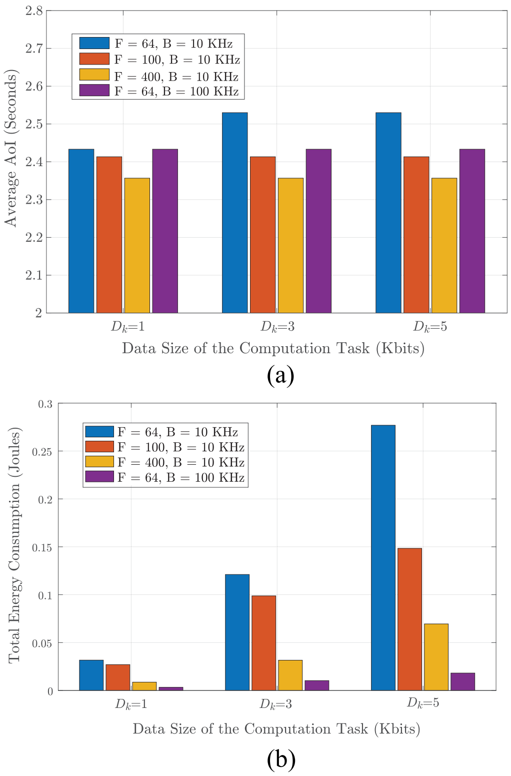
devices is different since the arrival of tasks is random. Similar to the analysis process presented in Fig. 2, when the device's task is scheduled at the beginning of a time slot, and offloaded and computed within the current time slot, AoI drops from a peak to the AoI of the scheduled task in the device.

E. Impact of Channel Quality

We present the impact of the number of AIRS reflective elements F and the bandwidth B on system performance and the time tradeoff between the offloading and computing in this section. The impact of communication quality determined by F and B on the AoI and energy consumption is shown in Fig. 10. Increasing the values of F or B can help improve information freshness and reduce energy consumption for a fixed-size computing task. This is because increasing F or B can increase the offloading rate, which is directly beneficial for reducing the offloading time. Under the limits of the processing deadline constraint (14), the computing time is thus sufficient, which is conducive to completing the offloading and computing in a timely and resource-conserving manner to improve the information freshness and reduce the energy consumption. The impact of D_k on the AoI and energy consumption has been indicated in Fig. 8.

F. Relation Between Device Offloading Power and HAP Computing Capability

We present and analyze the impact of the number of AIRS element F, HAP computing capability $\bar{f}$, and maximum offloading power of device $P_{\max}$ on the energy consumption in Fig. 11, which reveals the relation between the communication and computing resources. We observe that increasing F greatly reduces the average offloading energy E_{off} and thus the total energy consumption E_{total}, but the impact of F on the average computing energy E_{cop} is not significant. When we keep F constant and decrease $\bar{f}$ by 10 times, E_{cop} drops dramatically while E_{total} and E_{off} increase. When we decrease $P_{\max}$ by 10 times, E_{total} and E_{off} increases while E_{cop} decreases slightly.



(a)

(b)

Fig. 10. Impact of the number of AIRS element F and bandwidth B under different values of the computing task size D_k on the objective. (a) Impact of F and B on the average AoI. (b) Impact of F and B on the total energy consumption.

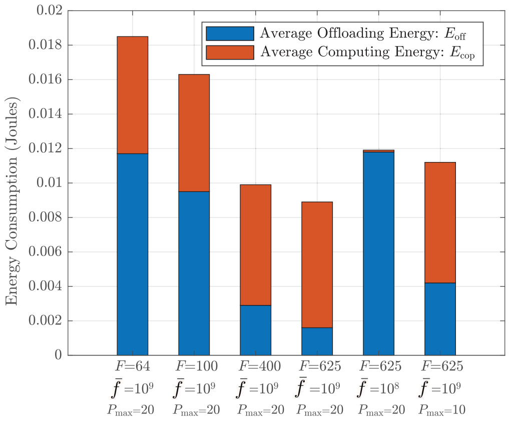


Fig. 11. Impact of parameters F, $\bar{f}$, and $P_{\max}$ on the tradeoff of communication energy and computation energy.

To further explore the relation between communication and computing resources and dig for more insights, we present the average values of energy, time, offloading power, and computing resources corresponding to communication and computing in Table III. p and f represent the average offloading power

TABLE III
AVERAGE VALUES OF ENERGY CONSUMPTION, TIME PROPORTION, AND RESOURCE VARIABLES UNDER DIFFERENT VALUES OF F, $\bar{f}$, AND $P_{\max}$

Parameter	E_{total} (J)	E_{off} (J)	E_{cop} (J)	p (W)	f (cycles/s)	T_{off} (s)	T_{cop} (s)
$F = 64, \bar{f} = 10^9, P_{\max} = 0.1$	0.0185	0.0117	0.0068	0.0254	2.6142×10^8	0.4617	0.0383
$F = 100, \bar{f} = 10^9, P_{\max} = 0.1$	0.0163	0.0095	0.0068	0.0207	2.6095×10^8	0.4617	0.0383
$F = 400, \bar{f} = 10^9, P_{\max} = 0.1$	0.0099	0.0029	0.0070	0.0064	2.6454×10^8	0.4622	0.0378
$F = 625, \bar{f} = 10^9, P_{\max} = 0.1$	0.0089	0.0016	0.0073	0.0036	2.6990×10^8	0.4629	0.0371
$F = 625, \bar{f} = 10^8, P_{\max} = 0.1$	0.0119	0.0118	1.0941×10^{-4}	0.0623	3.3076×10^7	0.1853	0.3023
$F = 625, \bar{f} = 10^8, P_{\max} = 0.01$	0.0112	0.0042	0.0070	0.0093	2.6383×10^8	0.4422	0.0379

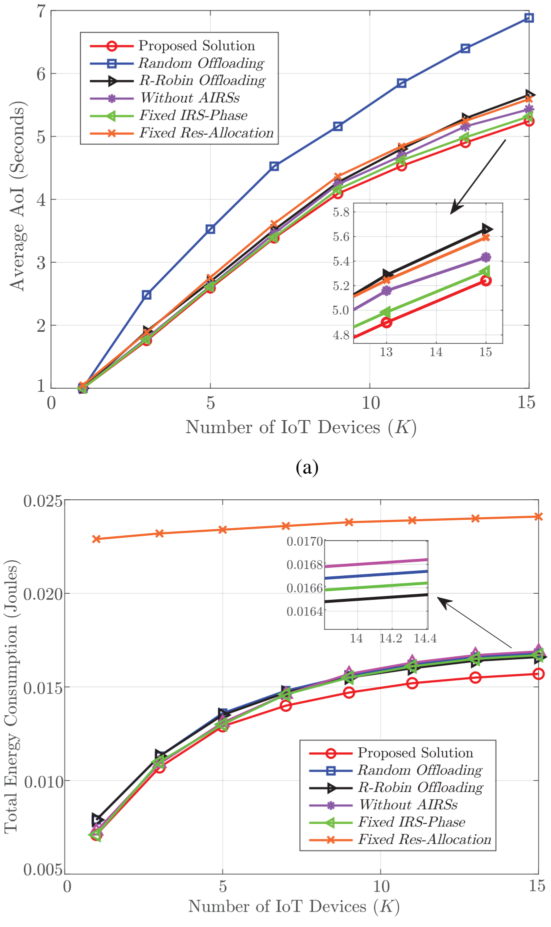
of the device and the average computing resources allocated by the HAP, respectively. We observe that when fixing $\bar{f}$ and increasing F, the proportion of the average computing time T_{cop} and average offloading time T_{off} is almost unchanged, the device tends to reduce p to save energy expenditure due to the enhancement of channel quality. However, when fixing F and reducing $\bar{f}$, the ratio of T_{cop} and T_{off} changes significantly. The proportion of T_{cop} increases due to the reduction in f, thereby E_{total} increases. Under the limitation of processing deadline, an increase in T_{cop} means a decrease in T_{off}, which is bound to increase the cost of p. Similarly, when fixing F and reducing $P_{\max}$, the reduction of p leads to an increase in offloading energy, thus f is increased to compensate for the energy loss caused by insufficient communication resources.

To sum up, communication and computing resources are mutually compensated. Increasing the communication or computing capabilities can help save energy, but the impact on the proportion of T_{off} and T_{cop} to δ is different. For communication, energy consumption is determined by F, p, and T_{off}. Enhancing the channel quality by increasing F can decrease p for saving energy but almost not affect T_{off}, thus has no impact on the allocation of f. However, the reduction in offloading capability, i.e., $P_{\max}$, will cause an increase in T_{off}. To ensure that the offloading and computing are completed within a given time, the HAP needs to increase f to save T_{cop}. For computing, energy consumption is determined by T_{cop} and f. T_{cop} is directly affected by the computing capability $\bar{f}$. The weakening of $\bar{f}$ increases T_{cop}, which forces the device to increase p to reduce T_{off} for satisfying (14).

G. Comparisons With Benchmarks

After understanding the influence mechanism of different parameters on the information freshness and energy consumption, we compare the proposed solution with five benchmarks in this section to evaluate the performance of the proposed solution. It is observed in Fig. 12 that both the average AoI AO and total energy consumption E_{total} increase with the number of devices K. This is because increasing K means that the time waiting for offloading becomes longer, the task volume increases, so the energy consumption increases. It is worth emphasizing that our proposed solution outperforms benchmarks in information freshness and energy consumption.

In terms of information freshness, as shown in Fig. 12(a), it is observed that *Random Offloading* has the worst information freshness performance. Even *R-Robin Offloading* performs better than *Random Offloading* but worse than other benchmarks. This indicates that the offloading scheme plays a dominant role in influencing AoI. The fixed AIRS phase shift, i.e., *Fixed IRS-Phase*, is closest to the performance of our proposed solution. Meanwhile, the optimization algorithm without the assistance of AIRSs, i.e., *Without AIRSs*, and the optimization algorithm of fixed resource allocation, i.e., *Fixed Res-Allocation*, are worse than *Fixed IRS-Phase*. In terms of energy consumption, it is observed in Fig. 12(b) that *Fixed Res-Allocation* has the worst performance, followed



(a)

(b)

Fig. 12. Comparisons between the proposed solution and benchmarks under different values of the number of devices K. (a) Comparisons of AO values. (b) Comparisons of E_{total} values.

by *Without AIRSs*. This is because unreasonable resource allocation and poor channel conditions inevitably waste communication and computing resources. Moreover, *R-Robin Offloading*, *Fixed IRS-Phase*, and *Random Offloading* all perform worse than the proposed solution. The above results indicate that the assistance of AIRSs, design of offloading, optimization of resource allocation, and design of IRS phase shift have different degrees of impact on the objective function and are essential for information freshness improvement and energy conservation in MEC networks.

VII. Conclusion

In this article, we have investigated the timely and reliable task offloading and computing assisted by multiple AIRSs for supporting energy-efficient and time-critical applications in MEC networks with poor links between devices and the edge server. We have proposed a low-complexity and efficient problem-solving framework to cope with the intractable formulated optimization problem that is mixed-integer and nonconvex. The proposed solution reveals the potential of the AIRS in improving information freshness and reducing energy consumption, explains the compensation relation between communication and computing resources, and provides design insights for task offloading and resource management. For future work, we will study the number of required AIRSs under specific requirements and explore the real-time offloading decision in AIRS-assisted MEC networks.

Appendix A
Proof of Theorem 2

This theorem is proved by contradiction. First, we denote the optimal solution to Problem $\mathrm{P}_{4.2a}$ by $\{t^{\mathrm{off}\,*}_{k,m}[t], e^*_{k,m}[t], f^*_m[t] \quad \forall k, m\}$. Without loss of optimality to Problem $\mathrm{P}_{4.2a}$, we can readily derive that the constraints in (40b) for all AIRSs and devices should be active, i.e.,

$$\bar{x}_{k,m}[t]\left(t^{\mathrm{off}\,*}_{k,m}[t] + \frac{D_k C_k}{f^*_m[t]}\right) = \delta \quad \forall k, m. \tag{41}$$

Otherwise, the objective function value can always be further decreased by decreasing $f^*_m[t]$ until the equality of (40b) holds, which also indicates the invariance of $t^{\mathrm{off}\,*}_{k,m}[t]$. Then, for the constraints in (39), in the optimal solution to Problem $\mathrm{P}_{4.2a}$, we assume that there exists a set of variables $\{t^{\mathrm{off}\,*}_{k,m}[t], e^*_{k,m}[t] \forall k, m\}$ such that

$$t^{\mathrm{off}\,*}_{k,m}[t] B \log_2\left(1 + \frac{e^*_{k,m}[t]|H_k[t]|^2}{t^{\mathrm{off}\,*}_{k,m}[t]\sigma^2}\right) > D_k. \tag{42}$$

Thus, we can always find another set of variables $\{t^{\mathrm{off}\,\star}_{k,m}[t], e^\star_{k,m}[t] \forall k, m\}$ which satisfy

$$t^{\mathrm{off}\,\star}_{k,m}[t] B \log_2\left(1 + \frac{e^\star_{k,m}[t]|H_k[t]|^2}{t^{\mathrm{off}\,\star}_{k,m}[t]\sigma^2}\right) = D_k. \tag{43}$$

For the set of variables $\{t^{\mathrm{off}\,\star}_{k,m}[t], e^\star_{k,m}[t] \forall k, m\}$, the constraints in (39) are active while the constraints in (40b) are inactive, thereby the objective function value of Problem $\mathrm{P}_{4.2a}$ can be further decreased. This contradicts the assumption. To sum up, in the optimal solution $\{t^{\mathrm{off}\,*}_{k,m}[t], e^*_{k,m}[t], f^*_m[t] \quad \forall k, m\}$ to Problem $\mathrm{P}_{4.2a}$, both the constraints in (40b) and (39) are active. Likewise, we can apply the same contradiction method to prove that in the optimal solution to Problem $\mathrm{P}_{4.2}$, the constraints in (14) are active for all devices and AIRSs. As a result, the optimal solution to Problem $\mathrm{P}_{4.2}$ can be obtained by solving the equivalent Problem $\mathrm{P}_{4.2a}$, leading to the desired result.

Appendix B
Convergence Analysis of Algorithm 4

First, given $p^{(i)}_{k,m}[t]$ and $f^{(i)}_m[t]$ in step 10 of Algorithm 4, the optimal solution of Problem $\mathrm{P}_{4.1}$ is obtained according to Algorithm 2. Thus, we have

$$\begin{aligned} & A^{(i)}_d\left(p^{(i)}_{k,m}[t], f^{(i)}_m[t], x^{(i)}_{k,m}[t]\right) \\ & \quad \leq A^{(i+1)}_d\left(p^{(i)}_{k,m}[t], f^{(i)}_m[t], x^{(i+1)}_{k,m}[t]\right) \end{aligned} \tag{44}$$

which can guarantee that $\mathrm{AO}^{(i)}[t] \geq \mathrm{AO}^{(i+1)}[t]$.

Second, for given $x^{(i)}_{k,m}[t]$ in step 12 of Algorithm 4, the total energy consumption E_{total} follows that:

$$\begin{aligned} & E^{(i)}_{\mathrm{total}}\left(x^{(i)}_{k,m}[t], p^{(i)}_{k,m}[t], f^{(i)}_m[t]\right) \\ & \quad \geq E^{(i+1)}_{\mathrm{total}}\left(x^{(i)}_{k,m}[t], p^{(i+1)}_{k,m}[t], f^{(i+1)}_m[t]\right) \end{aligned} \tag{45}$$

which holds since Problem $\mathrm{P}_{4.2}$ is solved optimally. The optimal phase shift can always be obtained via Lemma 1.

Based on the above analysis, we have

$$\begin{aligned} & \mathrm{Obj}^{(i)}\left(x^{(i)}_{k,m}[t], p^{(i)}_{k,m}[t], f^{(i)}_m[t]\right) \\ & \quad \geq \mathrm{Obj}^{(i+1)}\left(x^{(i+1)}_{k,m}[t], p^{(i+1)}_{k,m}[t], f^{(i+1)}_m[t]\right) \end{aligned} \tag{46}$$

where $\mathrm{Obj} = \mathrm{AO} + E_{\mathrm{total}}$ indicates the objective function value of Problem P_1, which means that Problem P_1 is monotonically nonincreasing as the iteration number increases. Furthermore, Obj is lower bounded by zero, the proposed algorithm can be guaranteed to converge to a stationary point.

References

[1] M. Vaezi et al., "Cellular, wide-area, and non-terrestrial IoT: A survey on 5G advances and the road toward 6G," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 2, pp. 1117–1174, 2nd Quart., 2022.

[2] F. Weidner et al., "A systematic review on the visualization of avatars and agents in AR & VR displayed using head-mounted displays," *IEEE Trans. Vis. Comput. Graph.*, vol. 29, no. 5, pp. 2596–2606, May 2023.

[3] M. Li, J. Gao, L. Zhao, and X. Shen, "Adaptive computing scheduling for edge-assisted autonomous driving," *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 5318–5331, Jan. 2021.

[4] M. Li, C. Chen, H. Wu, X. Guan, and X. Shen, "Age-of-information aware scheduling for edge-assisted industrial wireless networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5562–5571, Aug. 2021.

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[6] B. Ai et al., "Feeder communication for integrated networks," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 20–27, Dec. 2020.

[7] X. Shen, J. Gao, W. Wu, M. Li, C. Zhou, and W. Zhuang, "Holistic network virtualization and pervasive network intelligence for 6G," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 1–30, 1st Quart., 2022.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

[8] Q. Wu and R. Zhang, "Joint active and passive beamforming optimization for intelligent reflecting surface assisted SWIPT under QoS constraints," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1735–1748, Aug. 2020.

[9] Y. Su, X. Pang, S. Chen, X. Jiang, N. Zhao, and F. R. Yu, "Spectrum and energy efficiency optimization in IRS-assisted UAV networks," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6489–6502, Oct. 2022.

[10] Z. Ma et al., "Modeling and analysis of MIMO multipath channels with aerial intelligent reflecting surface," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 10, pp. 3027–3040, Oct. 2022.

[11] C. Wang et al., "Covert communication assisted by UAV-IRS," *IEEE Trans. Commun.*, vol. 71, no. 1, pp. 357–369, Jan. 2023.

[12] X. Zhang, M. Peng, and C. Liu, "Impacts of antenna downtilt and Backhaul connectivity on the UAV-enabled heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 22, no. 6, pp. 4057–4073, Jun. 2023.

[13] M. Samir, M. Elhattab, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Optimizing age of information through aerial reconfigurable intelligent surfaces: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3978–3983, Apr. 2021.

[14] H. Pan, Y. Liu, G. Sun, J. Fan, S. Liang, and C. Yuen, "Joint power and 3D trajectory optimization for UAV-enabled wireless powered communication networks with obstacles," *IEEE Trans. Commun.*, vol. 71, no. 4, pp. 2364–2380, Apr. 2023.

[15] M. A. Abd-Elmagid, N. Pappas, and H. S. Dhillon, "On the role of age of information in the Internet of Things," *IEEE Commun. Mag.*, vol. 57, no. 12, pp. 72–77, Dec. 2019.

[16] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1183–1210, May 2021.

[17] X. Chen et al., "Information freshness-aware task offloading in air-ground integrated edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 243–258, Jan. 2022.

[18] M. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao, and X. Shen, "Energy-efficient UAV-assisted mobile edge computing: Resource allocation and trajectory optimization," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3424–3438, Mar. 2020.

[19] J. Xie, Y. Jia, W. Wen, Z. Chen, and L. Liang, "Dynamic D2D multihop offloading in multi-access edge computing from the perspective of learning theory in games," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 1, pp. 305–318, Mar. 2023.

[20] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[21] F. Wang, J. Xu, and Z. Ding, "Multi-antenna NOMA for computation offloading in multiuser mobile edge computing systems," *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 2450–2463, Mar. 2019.

[22] A. Muhammad, M. Elhattab, M. A. Arfaoui, A. Al-Hilo, and C. Assi, "Age of information optimization in RIS-assisted wireless networks," *IEEE Trans. Netw. Service Manag.*, vol. 21, no. 1, pp. 925–938, Feb. 2024.

[23] W. Lyu, Y. Xiu, S. Yang, P. L. Yeoh, Y. Li, and Z. Zhang, "Weighted sum age of information minimization in wireless networks with aerial IRS," *IEEE Trans. Veh. Technol.*, vol. 72, no. 4, pp. 5390–5394, Apr. 2023.

[24] W. Jiang, B. Ai, M. Li, W. Wu, and X. Shen, "Average age-of-information minimization in aerial IRS-assisted data delivery," *IEEE Internet Things J.*, vol. 10, no. 17, pp. 15133–15146, Sep. 2023.

[25] W. Jiang, B. Ai, J. Cheng, Y. Lin, and G. Zhang, "Sum of age-of-information minimization in aerial IRSs assisted wireless networks," *IEEE Commun. Lett.*, vol. 27, no. 5, pp. 1377–1381, May 2023.

[26] Z. Chu et al., "Utility maximization for IRS assisted wireless powered mobile edge computing and caching (WP-MECC) networks," *IEEE Trans. Commun.*, vol. 71, no. 1, pp. 457–472, Jan. 2023.

[27] Z. Wang, Y. Wei, Z. Feng, F. R. Yu, and Z. Han, "Resource management and reflection optimization for intelligent reflecting surface assisted multi-access edge computing using deep reinforcement learning," *IEEE Trans. Wireless Commun.*, vol. 22, no. 2, pp. 1175–1186, Feb. 2023.

[28] T. Bai, C. Pan, Y. Deng, M. Elkashlan, A. Nallanathan, and L. Hanzo, "Latency minimization for intelligent reflecting surface aided mobile edge computing," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2666–2682, Nov. 2020.

[29] S. Xu, Y. Du, J. Liu, and J. Li, "Intelligent reflecting surface based backscatter communication for data offloading," *IEEE Trans. Commun.*, vol. 70, no. 6, pp. 4211–4221, Jun. 2022.

[30] G. Chen, Q. Wu, W. Chen, D. W. K. Ng, and L. Hanzo, "IRS-aided wireless powered MEC systems: TDMA or NOMA for computation offloading?" *IEEE Trans. Wireless Commun.*, vol. 22, no. 2, pp. 1201–1218, Feb. 2023.

[31] L. Guo, X. Sun, W. Zhang, Z. Li, and Q. Yu, "Small aerial target detection using trajectory hypothesis and verification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, May 2023, Art. no. 5609314.

[32] Q. Wu and R. Zhang, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 106–112, Jan. 2020.

[33] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu, "An efficient *K*-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[34] "Enhanced LTE support for aerial vehicles," 3GPP, Sophia Antipolis, France, 3GPP Rep. TR-36.777, 2018. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/36_series/36.777

[35] W. Wu, N. Cheng, N. Zhang, P. Yang, W. Zhuang, and X. Shen, "Fast mmwave beam alignment via correlated bandit learning," *IEEE Trans. Wireless Commun.*, vol. 18, no. 12, pp. 5894–5908, Dec. 2019.

[36] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.

[37] Q. Wu and R. Zhang, "Beamforming optimization for wireless network aided by intelligent reflecting surface with discrete phase shifts," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1838–1851, Mar. 2020.

[38] X. Huang, W. Wu, S. Hu, M. Li, C. Zhou, and X. S. Shen, "Digital twin based user-centric resource management for multicast short video streaming," *IEEE J. Sel. Top. Signal Process.*, early access, Dec. 18, 2023, doi: 10.1109/JSTSP.2023.3343626.

[39] X. Wang, Z. Fei, J. A. Zhang, J. Huang, and J. Yuan, "Constrained utility maximization in dual-functional radar-communication multi-UAV networks," *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2660–2672, Apr. 2021.

[40] K. Shen and W. Yu, "Distributed pricing-based user association for downlink heterogeneous cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1100–1113, Jun. 2014.

[41] K. R. Apt and A. Witzel, "A generic approach to coalition formation," *Int. Game theory Rev.*, vol. 11, no. 3, pp. 347–367, 2009.

[42] W. Wu et al., "Split learning over wireless networks: Parallel design and resource management," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 1051–1066, Apr. 2023.

[43] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ., 2004.

[44] F. Cheng et al., "UAV trajectory optimization for data offloading at the edge of multiple cells," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6732–6736, Jul. 2018.

[45] M. Grant and S. Boyd. "CVX: MATLAB software for disciplined convex programming, version 2.1." Mar. 2014. [Online]. Available: http://cvxr.com/cvx

Wenwen Jiang (Graduate Student Member, IEEE) received the B.E. degree in communication engineering from Shandong Normal University, Jinan, China, in 2018. She is currently pursuing the Ph.D. degree with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China.

She was a visiting Ph.D. student with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, from 2022 to 2023. Her research interests include UAV communications, IRS-assisted communications, and mobile edge computing.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.

Bo Ai (Fellow, IEEE) received the M.S. and Ph.D. degrees from Xidian University, Xi'an, China, in 2002 and 2004, respectively.

He was with Tsinghua University, Beijing, China, where he was an Excellent Postdoctoral Research Fellow in 2007. He was a Visiting Professor with the Electrical Engineering Department, Stanford University, Stanford, CA, USA, in 2015. He is currently a Full Professor with Beijing Jiaotong University, Beijing, China, where he is the Dean of the School of Electronic and Information Engineering, the Deputy Director of the State Key Laboratory of Rail Traffic Control and Safety, and the Deputy Director of the International Joint Research Center. He is one of the Directors for Beijing Urban Rail Operation Control System International Science and Technology Cooperation Base, Beijing, and the Backbone Member of the Innovative Engineering based jointly granted by the Chinese Ministry of Education and the State Administration of Foreign Experts Affairs. He is the Research Team Leader of 26 national projects. He holds 26 invention patents. His research interests include the research and applications of channel measurement and channel modeling and dedicated mobile communications for rail traffic systems. He has authored or coauthored eight books and authored over 300 academic research articles in his research area.

Dr. Ai has won some important scientific research prizes. Five papers have been the ESI Highly Cited Paper. He has been notified by the Council of Canadian Academies that based on the Scopus database, he has been listed as one of the Top 1% Authors in his field all over the world. He has also been feature interviewed by the IET *Electronics Letters*. He received the Distinguished Youth Foundation and Excellent Youth Foundation from the National Natural Science Foundation of China, the Qiushi Outstanding Youth Award by the Hong Kong Qiushi Foundation, the New Century Talents by the Chinese Ministry of Education, the Zhan Tianyou Railway Science and Technology Award by the Chinese Ministry of Railways, and the Science and Technology New Star by the Beijing Municipal Science and Technology Commission. He is an IEEE VTS Beijing Chapter Vice Chair and an IEEE BTS Xi' an Chapter Chair. He was a Co-Chair or a Session/Track Chair of many international conferences. He is an Associate Editor of the IEEE ANTENNAS AND WIRELESS PROPAGATION LETTERS and the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, and an Editorial Committee Member of the *Wireless Personal Communications* Journal. He is the Lead Guest Editor of Special Issues on the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE ANTENNAS AND PROPAGATIONS LETTERS, and the *International Journal on Antennas and Propagations*. He is a Fellow of The Institution of Engineering and Technology and an IEEE VTS Distinguished Lecturer.

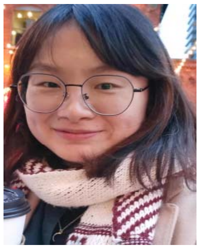
Mushu Li (Member, IEEE) received the M.A.Sc. degree from Toronto Metropolitan University, Toronto, ON, Canada, in 2017, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2021.

She is currently a Postdoctoral Fellow with Toronto Metropolitan University. She was a Postdoctoral Fellow with the University of Waterloo from 2021 to 2022. Her research interests include mobile edge computing, the system optimization in wireless networks, and machine learning-assisted network management.

Dr. Li was the recipient of Natural Science and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship in 2022, the NSERC Canada Graduate Scholarship in 2018, and the Ontario Graduate Scholarship in 2015 and 2016.

Wen Wu (Senior Member, IEEE) received the B.E. degree in information engineering from South China University of Technology, Guangzhou, China, in 2012, the M.E. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 2015, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2019.

He was a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Waterloo. He is currently an Associate Researcher with the Frontier Research Center, Peng Cheng Laboratory, Shenzhen, China. His research interests include 6G networks, network intelligence, and network virtualization.



Yingying Pei (Graduate Student Member, IEEE) received the B.E. degree from Huazhong University of Science and Technology, Wuhan, China, in 2014, and the M.Sc. degree from the University of Macau, Macau, China, in 2019. She is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada.

Her current research interests include network resource management for extended reality services.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular networks. Dr. Shen received the "West Lake Friendship Award" from Zhejiang Province in 2023, the President's Excellence in Research from the University of Waterloo in 2022, the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory in 2021, the R.A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society (ComSoc), and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and the AHSN Technical Committee in 2013. He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada. He serves/served as the General Chair for the 6G Global Conference'23 and ACM Mobihoc'15, Technical Program Committee Chair/Co-Chair for IEEE Globecom'24, 16, and 07, IEEE Infocom'14, and IEEE VTC'10 Fall, and the Chair for the IEEE ComSoc Technical Committee on Wireless Communications. He is the President of the IEEE ComSoc. He was the Vice President for Technical and Educational Activities, the Vice President for Publications, the Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and the Member of IEEE Fellow Selection Committee of the ComSoc. He served as the Editor-in-Chief for the IEEE INTERNET OF THINGS JOURNAL, *IEEE Network*, and *IET Communications*. He is a Registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Member, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:24:25 UTC from IEEE Xplore. Restrictions apply.