Air-Ground Collaborative Mobile Edge Computing: Architecture, Challenges, and Opportunities

Qin Zhen\textsuperscript{1,2}, He Shoushuai\textsuperscript{1}, Wang Hai\textsuperscript{1,\textsuperscript{*}}, Qu Yuben\textsuperscript{3,4}, Dai Haipeng\textsuperscript{5}, Xiong Fei\textsuperscript{6}, Wei Zhenhua\textsuperscript{7}, Li Hailong\textsuperscript{7}

\textsuperscript{1}College of Communications Engineering, Army Engineering University of PLA, Nanjing 210000, China
\textsuperscript{2}Department of Information and Communication, Noncommissioned Officer Academy of PAP, Hangzhou 310000, China
\textsuperscript{3}Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space, Ministry of Industry and Information Technology, Nanjing 211106, China
\textsuperscript{4}College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
\textsuperscript{5}State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
\textsuperscript{6}CMC Political and Law Commission, Beijing, 100120, China
\textsuperscript{7}Xi’an Research Institute of High Technology, Xi’an 710025, China
\textsuperscript{*}The corresponding author, email: haiwang@aeu.edu.cn

Cite as: Z. Qin, S. He, \textit{et al.}, “Air-ground collaborative mobile edge computing: Architecture, challenges, and opportunities,” China Communications, 2024, vol. 21, no. 5, pp. 1-16. DOI: 10.23919/JCC.ea.2021-0669.202401

Abstract: By pushing computation, cache, and network control to the edge, mobile edge computing (MEC) is expected to play a leading role in fifth generation (5G) and future sixth generation (6G). Nevertheless, facing ubiquitous fast-growing computational demands, it is impossible for a single MEC paradigm to effectively support high-quality intelligent services at end user equipments (UEs). To address this issue, we propose an air-ground collaborative MEC (AGC-MEC) architecture in this article. The proposed AGC-MEC integrates all potentially available MEC servers within air and ground in the envisioned 6G, by a variety of collaborative ways to provide computation services at their best for UEs. Firstly, we introduce the AGC-MEC architecture and elaborate three typical use cases. Then, we discuss four main challenges in the AGC-MEC as well as their potential solutions. Next, we conduct a case study of collaborative service placement for AGC-MEC to validate the effectiveness of the proposed collaborative service placement strategy. Finally, we highlight several potential research directions of the AGC-MEC.

Keywords: air-ground; architecture; collaborative; mobile edge computing

I. INTRODUCTION

Fifth-generation (5G) network has been deployed worldwide and commercially available in 2020, which offers many more functions than previous generations [1]. However, with the advancement of smart devices and Internet of Things (IoT) technology, as well as diversified applications (e.g., smart city, mobile augmented reality, face recognition, and autonomous driving), 5G networks cannot completely meet future rapidly growing traffic demands. Accordingly, sixth-generation (6G) have attracted increasing attention from both industry and academia, which will be transformative and revolutionize the wireless evolution form. 6G network is expected to effectively support high-quality services and unlimited connectivity for a large number of intelligent devices [2, 3]. Meanwhile, it brings great challenges to the computing power of centralized data center and intelligent terminals. The traditional cloud computing cannot meet...
the requirements of massive data processing, and computing power will be transferred from the network core to the network edge. Mobile edge computing (MEC) is an emerging computing paradigm that can push mobile computing, cache, and network control to the edge in the close proximity of mobile user equipments (UEs) [4]. MEC is envisioned to play a leading role in 6G by operating as an intermediate layer that provides fast and localized data processing for many critical and resource-constrained applications [5].

The concept of MEC was firstly proposed by the European Telecommunications Standard Institute (ETSI) in 2014. MEC has attracted increasing attention in academia, which is regarded as one of the most cutting-edge technologies in the wireless communications field. Many scholars have actively explored the academic research of MEC in the fields of 5G and Beyond 5G (B5G) communications, IoT communications, unmanned aerial vehicle (UAV) communications, and artificial intelligence (AI). Furthermore, in China’s 14th Five-Year Plan, it is mentioned that “Develop both cloud services and edge computing services and foster the Internet of Vehicles, medical IoT, and home IoT industries”. Now, MEC is widely agreed to be a key technology for realizing various visions for next generation Internet [4].

With the help of MEC, computation-intensive and latency-sensitive tasks can be offloaded for remote execution, which can enhance the computing ability, and reduce energy consumption and latency. MEC servers are usually deployed in a fixed fashion at the ground base stations (BSs), wireless access points (APs), and roadside units (RSUs). Nevertheless, such formed traditional terrestrial infrastructure-based MEC system has its limitations, which may not work in many critical applications, such as military, emergency relief, and disaster response. In addition, since terrestrial MEC servers lacks mobility, it cannot meet the computation and connectivity demands with the spatio-temporal dynamics. In contrast to the terrestrial MEC network, due to the Line-of-Sight (LoS) links, flexible deployment, and maneuverability, the aerial MEC network consisting of UAVs, airships, and balloons equipped with MEC servers might compensate those weaknesses [6, 7]. The aerial MEC network is also faced with many challenges such as the limited battery life. As a result, considering intelligent endogenous and ubiquitous computational power requirements of 6G networks, it is impossible for a single MEC paradigm to accomplish such a difficult task.

To promote the development of 6G networks, we propose an air-ground collaborative MEC (AGC-MEC) architecture, which actively explores the complementary integration of computational powers from both air and ground segments to provide intelligent deployment and management of different MEC paradigms. The novelty of the proposed AGC-MEC mainly lies in that it involves all potentially available MEC servers in the air-ground integrated networks. Specifically, the AGC-MEC architecture comprises a two-layer networking architecture: aerial MEC and terrestrial MEC. In the aerial MEC, airships and balloons are deployed as (near) static aerial MEC servers, while UAVs as mobile aerial MEC servers. In the terrestrial MEC, BSs/APs/RSUs are deployed as static terrestrial MEC servers, while vehicles as mobile terrestrial MEC servers. Meanwhile, powerful mobile users can be used as opportunistic mobile ground MEC servers. Through diverse collaborations between the air and ground, the AGC-MEC architecture is more flexible than any single terrestrial MEC network and more powerful than any single aerial MEC network.

There exist several studies that investigate air-ground integrated MEC network. They can be roughly divided into two categories: one considers aerial MEC servers only [8–12] and the other considers both aerial and terrestrial MEC servers [13–17]. However, all the aforementioned studies only consider partial MEC paradigms in the air-ground integrated network. For example, [8–11] consider the UAV-enabled MEC solely, [12] considers the balloon-enabled MEC solely, while [13, 14] consider static terrestrial MEC servers and UAVs, [15] considers mobile ground vehicles and UAVs, and [16, 17] additionally consider the fixed ground BS based on [15]. To the best of our knowledge, there is no architecture that comprehensively considers integrating all potentially available MEC servers within air and ground. In contrast, our proposed AGC-MEC involves all potentially available MEC servers (both static and mobile) within air and ground in the context of future 6G networks, by a variety of collaborative ways to provide computation services at their best for UEs. Furthermore, AGC-MEC considers how to enable different MEC servers collaborate effectively, rather than merely integrate them.

In the rest of this article, we first introduce the re-
related work in Section II. We present the proposed AGC-MEC architecture and elaborate three typical use cases to illustrate its potential values in Section III. Then, we analyze several main technical challenges of the AGC-MEC in Section IV. Next, we conduct a case study to evaluate the performance of the AGC-MEC in Section V, while we discuss three potential research directions in Section VI. Finally, we draw conclusions in Section VII.

II. RELATED WORK

2.1 Traditional Terrestrial MEC

In recent years, there has been many literatures devoted to study on traditional terrestrial MEC network. The original definition of MEC referred to the use of BSs for offloading computing tasks from mobile users [4]. Nowadays, the terrestrial MEC servers are extended to ground APs and RSUs, etc., which are usually deployed in a fixed fashion. Chen et al. investigated a software defined ultra-dense network, which included many densely deployed BSs equipped with MEC servers. In order to save the battery life and reduce the delay, UEs can offload tasks to BSs or perform locally [18]. Tran et al. studied a MEC enabled multi-cell wireless network, which included multi-cell and BSs equipped with MEC servers that can provide computing services for mobile users [19]. Zhang et al. advocated that the integration of computation, cache, and communication at the network edge, e.g., APs and RSUs, provided an effective framework to address the data processing, aggregation, and acquisition challenges for intelligent internet of vehicles [20]. In fact, traditional terrestrial MEC system has its limitations, which may not work in many critical applications, such as emergency relief, military, and disaster response. In addition, terrestrial MEC servers deployed in a fixed fashion may not meet the computation and connectivity demands with the spatio-temporal dynamics. Fortunately, aerial MEC servers, e.g., UAVs, airships, and balloons, might compensate those weaknesses due to the flexible deployment, LoS links, and maneuverability.

2.2 Air-Assisted MEC

With the help of aerial MEC servers, air-assisted MEC network can be regarded as a promising technology to address problems and challenges caused by traditional terrestrial MEC system. There exist several studies that investigate air-assisted MEC network. They can be roughly divided into two categories: one considers aerial MEC servers only and the other considers both aerial and terrestrial MEC servers.

For the former one, Zhou et al. introduced three UAV-enabled MEC architectures, which can improve computation performance and reduce execution latency by integrating UAV into MEC networks [8]. Qu et al. investigated service provisioning for UAV-enabled MEC, where multiple rotary-wing UAVs provided personalized computing service for ground users [9]. Cheng et al. proposed a novel air-ground integrated mobile edge network (AGMEN), where UAVs can be flexibly deployed and scheduled, and assisted the computing, caching, and communication of the edge network [10]. You et al. studied a flying ad hoc networks (FANETs) surveillance system, where UAVs was used to collect data over a wide range of areas. They selected cluster head UAVs with powerful computing resources as aerial MEC servers to perform computational tasks [11]. Wang et al. considered a MEC-enabled high-altitude balloons (HABs) network, where a set of balloons can be used as aerial MEC servers to help ground users to offload computation-intensive tasks [12]. Compared with UAVs, balloons can be equipped with powerful energy and computing resources, which can continuously hover to provide computing services to users.

For the latter one, Zhao et al. investigated a UAV-assisted MEC network in industrial production, where multiple UAVs and a ground MEC server cooperated to provide computing services for industrial sensing vehicles [13]. Peng et al. considered a UAV-assisted vehicular networks, where the UAVs and macro eNodeB can be equipped with MEC servers to help vehicles to complete heterogeneous computation-intensive and time-sensitive tasks [14]. Zhou et al. proposed an air-ground integrated MEC framework, where ground vehicles and UAVs can be envisaged as supplementary MEC servers for efficient service provisioning [15]. Jiang et al. studied a hybrid MEC platform, where UEs can offload their computation tasks to UAVs, vehicles, and ground BSs, all equipped with edge computing servers [16]. Jiang et al. proposed a heterogeneous MEC (H-MEC) architecture, which aimed to address the key challenges of the H-MEC architec-
Table 1. A comparison of different MEC paradigms in AGC-MEC.

| MEC Paradigms | UAV | Airship/Balloon | BS/AP/RSU | Vehicle | Powerful Mobile User |
|---------------|-----|-----------------|-----------|---------|----------------------|
| Location      | Air | Air             | Ground    | Ground  | Ground               |
| Cost          | Small | High            | High      | Medium  | Small                |
| Availability  | Medium | High            | High      | Medium  | Low                  |
| Reliability   | Low | High            | High      | Medium  | Low                  |
| Mobility      | Mobile | Quasi-Stationary | Static   | Mobile  | Mobile               |
| (3D)          | (3D)    | (2D and Restricted) | (2D and Very Restricted) |
| Energy Supply | Poor | Medium          | Abundant  | Medium  | Poor                 |
| Coverage      | Medium | Large            | Large     | Medium  | Small                |
| Computation Power | Weak~Medium | Medium~Strong           | Strong    | Medium  | Very Weak            |
| Communication Ability | Weak~Medium | Medium~Strong           | Strong    | Medium  | Very Weak            |
| Storage Capacity | Weak~Medium | Medium~Strong           | Strong    | Medium  | Very Weak            |

In dynamic environments using AI-based solutions [17].

However, all the aforementioned studies only consider partial MEC paradigms in the air-ground integrated network. To the best of our knowledge, there is no architecture that comprehensively considers integrating all potentially available MEC servers within air and ground, by a variety of collaborative ways to provide computation services at their best for UEs.

III. AIR-GROUND COLLABORATIVE MOBILE EDGE COMPUTING (AGC-MEC)

In this section, we propose the AGC-MEC architecture, which integrates all potentially available MEC servers within air and ground by a variety of collaborative ways to deliver computation services at their best for UEs. Firstly, we introduce the AGC-MEC architecture, which consists of aerial and terrestrial MEC networks. Secondly, in order to show the potential of the AGC-MEC, we elaborate three typical use cases.

3.1 AGC-MEC Overview

Figure 1(a) illustrates the conceptual architecture of the proposed AGC-MEC architecture, which comprises a two-layer networking architecture following the general air-ground integrated networks: aerial MEC and terrestrial MEC. Specifically, the aerial MEC network is mainly composed of UAVs, airships, and balloons, while the terrestrial MEC network is generally made up of BSs, APs, RSUs, vehicles, and mobile users, all of which are equipped with edge computing servers. In the aerial MEC, airships and balloons are deployed as (near) static aerial MEC servers and UAVs as mobile aerial MEC servers. In the terrestrial MEC, there are three types of MEC servers. Specifically, BSs/APs/RSUs and vehicles are employed as static and mobile terrestrial MEC servers, while some powerful mobile users can be used as opportunistic mobile terrestrial MEC servers.

Different MEC paradigms have their particular features, which are partially overlapping but also complementary. We present a comprehensive comparison of different MEC paradigms in the AGC-MEC architecture in Table 1. MEC servers are usually deployed in a fixed fashion at the BSs, APs, RSUs, airships, and balloons. Despite their high availability and reliability, they also bring higher costs than UAVs, vehicles, and powerful mobile user. Due to mobility and flexibility, UAVs and vehicles can be deployed quickly on demand. UAVs move much faster than vehicles, but with less computation, communication, and storage resources. On the contrary, vehicles move slower but hold more resources. In fact, BSs, APs, RSUs, airships, and balloons have the most available resources. Furthermore, the coverage of fixed MEC server is large but remains unchanged. They cannot exploit its mobility to move closer to UEs with computation-intensive tasks. In contrast, UAVs, vehicles, and powerful mobile users have limited coverage and energy, but can move close to UEs to provide low-latency services and communication. By collaborating different MEC paradigms of air and ground, the AGC-MEC architecture aims to manage and control heterogeneous network resources smartly to meet computing demands.

As shown in Figure 2, we classify the collaborations of the AGC-MEC from two perspectives. From the perspective of the planes where servers are located, the collaboration of MEC servers can be divided into two types: vertical collaboration and horizontal collaboration. Specifically, vertical collaboration rep-
Collaborations in AGC-MEC

| Aerial MEC | Collaborations in AGC-MEC | Terrestrial MEC |
|------------|---------------------------|-----------------|
| Near Static| Airship                   | Static          |
|            | Balloon                   | Mobile          |
| Mobile     | UAV                       | Power            |
|            |                           | User            |

(a) Collaborations between air and ground.

(b) Typical use cases.

**Figure 1. AGC-MEC architecture.**

represents collaboration possibilities among various layers, which can be divided into four categories as follows: A) static-air and static-ground collaboration; B) static-air and mobile-ground collaboration; C) mobile-air and static-ground collaboration; D) mobile-air and mobile-ground collaboration. The horizontal collaboration represents collaboration possibilities among the same layer, which can be divided into two categories as follows: E) static-air and mobile-air collaboration; F) static-ground and mobile-ground collaboration. For example, airship/balloon has strong computing power and rich resources, which can help UAVs deal with complex computing tasks. In the cooperative driving scenario, BSs and RSUs can serve as coordinators and computing servers for vehicles [21].

From the perspective of collaboration work mode, the collaboration of MEC servers can be divided into four types: parallel collaboration mode, serial collabo-
Parallel collaboration is one of the most commonly used collaboration work modes in MEC networks. It refers to the collaboration between different MEC paradigms to provide personalized computing services for different UEs, so as to realize the parallel processing of all kinds of computing tasks. UEs can select a suitable MEC server to offload their computation tasks according to their task requirements and resource consumption. For example, Shang et al. considered an air-to-ground integrated MEC network, in which UAVs and ground computational APs cooperated to provide computing resources for multiple UEs [22].

Serial collaboration refers to the collaboration of different MEC paradigms to provide different computing services to the same user, so that they can process computing tasks serially. Specifically, in order to make full use of the different advantages of various MEC servers, complex tasks can be decomposed, and then different MEC servers are used to process different subtasks. For example, visual target tracking tasks have high reasoning accuracy and strict delay requirements, which require excessive computing resources. Due to the limited computing resources and energy budget of UAVs, Yang et al. considered offloading tasks to ground MEC server to further improve the reasoning accuracy. Specifically, UAVs are embedded with the lower layer of the pre-trained convolutional neural network (CNN) model, while the ground MEC server has rich computing resources to deal with the higher layer of the CNN model [23].

Supporting collaboration is one of the important collaboration modes in MEC networks. It refers to that various MEC servers cooperate to provide computing services for UEs while using their respective advantages to provide help to each other. For example, due to high mobility and flexible deployment, UAVs can act as relays to transmit the UE’s computing tasks to ground MEC servers [24, 25]. Likewise, since ground MEC server has sufficient energy and powerful computing capabilities, it can provide charging services for UAVs with limited energy [26, 27].

Leadership collaboration is a special collaboration mode in MEC networks. It refers to that the MEC server with rich resource can act as the coordinator of MEC networks to guide other MEC servers to perform computing tasks (e.g., offloading decision, trajectory design, and resource allocation). For instance, Liu et al. considered a cooperative UAV-enabled MEC network, which mainly included three layers: system layer, UAV layer, and UE layer. Specifically, at the system layer, the BS equipped with MEC server is responsible for collecting network information (e.g.,
channel status information, computing capabilities of UAVs, and task requirements) and allocating computing/communication resources for UAVs and UEs [28]. The BS acts as a coordinator and makes full use of the global information for coordination control, while the UAV is responsible for specific computing tasks according to the instructions of the BS.

3.2 Typical Use Cases

As illustrated in Figure 1(b), the proposed AGC-MEC architecture is envisioned to be useful particularly in several applications as follows.

3.2.1 Unexpected Latency-Sensitive Applications

Unexpected latency-sensitive applications mainly refer to applications with sudden scenario and latency-sensitive tasks. With the development of IoTs and various mobile applications, more and more data is generated at the edge of the network. It is a general trend to process and analyze the data in real time widely at the network edge, which needs strong computing power support [29]. By effectively collaborating diverse MEC paradigms, the AGC-MEC architecture can provide efficient and flexible computing services at the edge to meet the demands of unexpected latency-sensitive applications. For example, intelligent transportation system (ITS) requires low-latency communication and high computation capabilities. Static ground RSUs equipped with MEC servers can process the local data, which not only reduces the burden of network transmission, but also speeds up the data processing. Nevertheless, there are situations where additional air and ground MEC servers are required to handle high traffic loads during extreme traffic congestion or unexpected weather conditions. According to congestion conditions and traffic events, UAVs can be dynamically deployed as mobile aerial MEC servers. In addition, MEC can play a significant role in connected healthcare systems by offering better insight of heterogeneous healthcare content to support affordable and quality patient care [30]. Ubiquitous collaborative MEC servers can help patients choose better advice from the right guardians in real time when some emergencies occur.

3.2.2 Temporary Spatio-Temporal Dynamic Applications

Temporary spatio-temporal dynamic applications mainly refer to applications with temporary scenario and spatio-temporal dynamic tasks. In the daily life, mobile UEs have obvious group effect and usually form different dense crowds with computing requests over time in special places, which results in a spatio-temporal dynamic computing demand. The AGC-MEC can deploy various MEC servers on demand to meet such a dynamic demand of UEs. One typical example scenario is the stadium in a major sport or concert event. There are a huge number of people gathered, who execute computation-intensive applications in their mobile phones such as Virtual Reality (VR) and online gaming. In this case, BSs may be overloaded and cannot support massive UEs. Fortunately, UAVs, airships, balloons, and vehicles can be temporarily deployed to collaborate with ground BSs to offload computation tasks and improve the user quality of experience (QoE). Other typical example scenario is tourist attractions and important transportation hubs [31]. For example, on October 3, 2019, Tiananmen received 2.68 million tourists throughout the day; the passenger flow reached the peak of that day at the time of flag lowering, which was 180 thousand. To meet the dynamic computing demands of tourists, UAVs can be deployed flexibly to collaborate with ground MEC systems during holidays.

3.2.3 Emergency and Military Applications

In some extreme cases such as emergency and military applications, the ground infrastructure-based MEC system may be destroyed partially or completely and thus cannot work properly. The AGC-MEC can assemble mobile MEC servers including UAVs, airships, balloons, and vehicles to assist the existing communication infrastructure, if any, more critically, in providing computing services. For instance, in the event of natural disasters and earthquakes, ground infrastructure may be damaged. The rescue crews may need mobile augmented reality to search the area, which needs a significant amount of computing resources. The AGC-MEC can provide the required computation resources to increase the search scope and speed up the rescue. Moreover, in military applications, there are massive computation-intensive reconnaissance tasks.
(e.g., estimating the locations and the dynamics of the hostile forces). For a large-scale reconnaissance area without infrastructure, UAVs can collaborate with ground vehicles to expand the reconnaissance scope, reduce cloud computing latency, and enhance the computing ability.

IV. CHALLENGES IN AGC-MEC

Due to the features of high mobility, heterogeneity, frequent inevitable air-ground interactions, and time-varying channel conditions, the AGC-MEC architecture is difficult in platform integration, network deployment, resource management, and intelligence realization. In this section, we discuss four challenges and their potential solutions.

4.1 Generic Computing Platform Integration

Generic computing platform integration is critical to ensure synergy between different MEC paradigms. The AGC-MEC involves different MEC paradigms, which exist in different segments and have distinct characteristics. MEC paradigms are connected to each other through a communication protocol and communicate with users and devices. However, different MEC paradigms have various communication protocols, communication links, and interfaces, which significantly limits interoperability [10]. Therefore, how to integrate the computing resources of various MEC paradigms and build a generic computing platform is our primary consideration.

Network Function Virtualization (NFV) is an emerging network technology, which moves the network function from the original special equipment to the general equipment. Specifically, it decouples network functions from specialized hardware, and can be leveraged to flexibly implement network functions as software instances in the network slices. Despite its great potential benefits, NFV is also faced with some problems to be solved. For example, under the virtualized network environment, which interfaces can be used privately, and which interfaces need to be standardized, these issues remain to be clarified.

4.2 On-Demand 3D Network Deployment

In the AGC-MEC architecture, a fundamental and critical issue is how to deploy air/ground MEC servers, which includes two aspects. For one thing, based on the demand, we need to determine when, where, what MEC paradigms, and how many MEC servers to deploy. For another thing, we need to determine how to collaborate optimize the trajectories of mobile air/ground MEC servers.

There are many challenges for the on-demand deployment of edge servers. First, due to the high mobility of users, heterogeneity of QoS requirements, and stochastic characteristics of wireless network, the deployment of MEC servers is exposed to extreme difficulties. Second, since AGC-MEC architecture involves the collaboration between air and ground, the MEC servers need to be deployed in 3D space. The critical design challenge is to adaptively adjust the trajectories to meet the dynamic computing task requirements. The existing work considers to use Deep Reinforcement Learning (DRL) algorithm, which can learn optimal placement policies and plan trajectories of mobile MEC servers intelligently [17]. However, DRL algorithm brings more computational cost than traditional methods.

4.3 Computing-Oriented Resource Allocation

Resource allocation is an important guarantee for effective collaboration of AGC-MEC architecture, which directly impacts network performance. In order to effectively complete computing tasks, the AGC-MEC architecture should carry out efficient, adaptive, and intelligent resource allocation. Air-ground collaborative resource allocation involves various resources, which can be mainly divided into the following two types: the communication resource including bandwidth and channels, and the computation resource including computing power and storage capacity. In particular, to provide personal services for different users, MEC servers need to store corresponding data including object databases, libraries and trained machine learning models, associated with services. Service provisioning is an essential and critical issue, i.e., how to determine where to store/place which MEC service to meet various computing service demands. Furthermore, jointly optimizing resource allocation, trajectory, and placement of MEC servers can obtain globally optimal performance that would be more useful in practice.

There are two main challenges to deal with the
above problem. First, the problem includes many decision variables to be jointly optimized, e.g., the continuous computation resource allocation and UAV trajectory variables, and integral task offloading and service placement variables. Therefore, the optimization problem is a non-convex mixed integer nonlinear programming (MINLP) problem and is hard to solve in general. Secondly, computing demands vary in time and space. In order to match resource provision with computing tasks, resource allocation needs to be adjusted dynamically in a real-time manner. In order to better adapt to the dynamic environment, the complicated network behaviors can be analyzed in real time, and the online algorithm can be used to flexibly adjust resource allocation.

4.4 Ubiquitous Edge Intelligence Realization

Edge intelligence (EI) is emerging as a promising key enabler for MEC to fulfill the vision of ubiquitous intelligence, which pushes intelligence to the network edge by running AI algorithms on edge devices. EI can be divided into two main types of technology: AI for edge and AI on edge [32]. The former focuses on utilizing AI algorithms to provide effective solutions for key problems in edge computing, and the latter focuses on realizing AI model training and inference on the edge. However, it is difficult to realize ubiquitous EI in the following aspects.

**AI for AGC-MEC:** It is devoted to provide better solutions to constrained optimization problems, e.g., trajectory optimization, resource allocation, network deployment, and real-time decision making. Despite its great potential benefits, utilizing AI algorithms is also faced with some problems to be solved. Firstly, since some MEC servers is resource-constrained, how to balance optimality and efficiency of the AGC-MEC architecture is a great challenge. Secondly, if we want to utilize AI algorithms to obtain solutions, the formulated optimization problem and mathematical model need to be restricted [32]. Therefore, the model establishment is a huge challenge.

**AI on AGC-MEC:** The computing and storage capacity of edge servers is far less than that of cloud servers, which cannot meet the needs of a large number of computing and storage resources for AI training. Fortunately, federated learning (FL) can make multiple resource-constrained end devices to collaboratively train effective learning models, which is an emerging distributed learning architecture [33]. However, it is challenging in learning-oriented training configuration, and energy efficient training strategies.

V. CASE STUDY: COLLABORATIVE SERVICE PROVISIONING FOR AGC-MEC

In this section, we conduct a case study of collaborative service provisioning for AGC-MEC to validate the effectiveness of the proposed collaborative service placement strategy. Firstly, we introduce the system model and problem formulation. Secondly, we describe the simulation settings and comparison algorithms. Finally, we show the simulation results.

5.1 System Model and Problem Formulation

5.1.1 Network Model

We consider an air-ground collaborative MEC network, which is composed of one UAV, one BS, and multiple UEs. The UAV and BS can collaborate with each other to provide various types of computing services for UEs. The computation-intensive tasks of UEs can be operated locally or be offloaded to the UAV or BS.

5.1.2 Latency Model

The latency is composed of two parts: computation time taken to execute tasks, communication time taken to offload tasks. The time for transmitting computation results is usually ignored. The computation time is calculated by the required number of CPU frequency cycles and allocated computing resources. The communication time is calculated by the input data size and transmission rate.

5.1.3 Energy Model

The energy consumption of the UE includes computation and communication energy consumption. The computation energy is related to the effective switched capacitance, local computing power and required number of CPU frequency cycles. The communication energy is related to the communication time and transmission power of the UE.
5.1.4 Problem Formulation

We aim to minimize UEs’ overall energy consumption by jointly optimizing service placement, task offloading, UAV trajectory, and computation resource allocation. The optimization problem can be formulated as:

- **Optimization objective**: the minimization of UEs’ overall energy consumption $F$.
- **Optimization variables**: task offloading $O$, service placement $P$, UAV trajectory $Q$, and computation resource allocation $R$.
- **Constraint 1**: storage capacities of the BS and UAV, which store corresponding data including object databases, trained machine learning models, and libraries, associated with services.
- **Constraint 2**: the restriction of maximum number of UEs associated with the BS and UAV.
- **Constraint 3**: the coverage area of the UAV.
- **Constraint 4**: the limitation of flight distance.
- **Constraint 5**: the delay requirements of UEs.
- **Constraint 6**: offloading condition, i.e., the required service need to be placed in the BS or UAV.

The optimization problem is a MINLP problem, which includes continuous computation resource allocation and UAV trajectory variables and integral task offloading and service placement variables. Therefore, this problem is NP-hard that is challenging to be solved [34].

5.1.5 Solution

To deal with this problem, we exploit alternating optimization techniques to propose a suboptimal solution with convergence guarantee. To be specific, we obtain the closed form of the optimal computation resource allocation and iteratively solve the task offloading and service placement subproblem by Branch and Bound (BnB), and UAV trajectory subproblem by successive convex approximation (SCA).

The alternating iteration optimization algorithm is shown in Algorithm 1. In Step 1, according to feasible initial solution $(O_0, P_0, Q_0, R_0)$, we calculate the initial objective function value $F_0$. Specifically, we take the diagonal of the region as the initial trajectory of the UAV. The initial feasible solution of task offloading and service placement is determined by random strategy. Meanwhile, according to the delay requirements, the minimum computing resources required by UEs are determined. In Step 2-Step 12, we decompose the original problem into three subproblems and iteratively solve three subproblems. Firstly, the task offloading and service placement subproblem can be regarded as a multi-dimensional multiple-choice knapsack problem (MMKP). It can be optimally solved by the BnB method via a standard optimization solver MOSEK (Step 3). Secondly, we use SCA algorithm to obtain the locally optimal solution of UAV trajectory subproblem (Step 4) [35, 36]. Thirdly, we obtain the closed form of the optimal computation resource allocation (Step 5). The core idea of solving the computation resource allocation subproblem is to calculate the minimum computation resource allocation variable to further save resources under the premise of satisfying the delay requirements. Then, we calculate the objective function value of this iteration (Step 6). Finally,

---

**Algorithm 1. Alternating iteration optimization algorithm.**

**Input:** Feasible initial solution $(O_0, P_0, Q_0, R_0)$, threshold $\delta$, iterative times $i = 1$, and maximum iterative times $i_{\text{max}} = 100$.

**Output:** Task offloading $O$, service placement $P$, UAV trajectory $Q$, and computation resource allocation $R$.

1: According to initial solution $(O_0, P_0, Q_0, R_0)$, calculate the initial objective function value $F_0$;
2: while $i < i_{\text{max}}$ do
3: Fixed UAV trajectory $Q_{i-1}$ and computing resource allocation $R_{i-1}$, obtain task offloading $O_i$ and service placement $P_i$ by BnB method;
4: Fixed task offloading $O_i$, service placement $P_i$, and computing resource allocation $R_{i-1}$, obtain UAV trajectory $Q_i$ by SCA method;
5: Fixed task offloading $O_i$, service placement $P_i$, and UAV trajectory $Q_i$, calculate optimal computing resource allocation $R_i$;
6: Calculate the objective function value of this iteration $F_i$;
7: $i \leftarrow i + 1$;
8: if $|F_{i-1} - F_{i-2}| / F_{i-2} \leq \delta$ then break;
9: end if
10: end while
11: Return $(O_{i-1}, P_{i-1}, Q_{i-1}, R_{i-1})$. 
we compare the objective function value of the last iteration to decide whether to stop the iteration or enter the next iteration (Step 7-Step 11).

5.2 Simulation Settings and Comparison Algorithms

This case considers an air-ground collaborative MEC network with a mobile UAV, a static BS and twenty UEs in a $200m \times 200m$ squared area. The UAV and BS are equipped with MEC servers and UEs are randomly distributed in the squared zone. We divide the whole mission period into 100 time slots, and the length of time slot is $1s$. Each UE generates a computation task constantly in each time slot. The number of required CPU cycles is in the range of $[10^8, 10^9]$ cycles and the size of input data is in the range of $[100, 1000]$ KB [35, 37]. We assume that the service population of UEs’ requests are produced according to the popular Zipf distribution with a skewness parameter value of 0.5 form a service set with size 30 [38]. The storage size required by each services is uniformly chosen from $[0.5, 1]$. We assume that the storage capacity of the BS is twice that of the UAV. The maximum transmission power of UEs equals to $0.1w$ and the bandwidth is $1MHz$. For computation energy consumption model, we assume the effective switched capacitance is $10^{27}$ [36]. UAV related parameters are set as follows: the maximum ground coverage radius is $100m$, maximum flying distance in a time slot is $30m$, maximum computation capacity is $[5, 8]GHz$, and the maximum number of UEs associated with UAV is $[3, 5]$. BS related parameters are set as follows: the maximum computation capacity is $[10, 15]GHz$ and the maximum number of UEs associated with BS is $[5, 8]$ [9].

To verify the performance of the proposed algorithm, we compare it with three algorithms as follows:

- Random: according to storage capacities of the BS or UAV, this algorithm places services randomly;
- Greedy: this algorithm uses a greedy strategy to select the service with the minimum required storage, which results in maximizing the number of services placed at the BS or UAV;
- Local: this algorithm executes tasks locally, which obtains an upper bound of UEs’ overall energy consumption.

5.3 Simulation Results

Figure 3 shows the convergence performance of four algorithms. It can be seen that after three iterations, UEs’ overall energy consumption of the proposed algorithm decreases from 274.61J to 94.04J, which means our algorithm converges quickly and has good performance. Random algorithm and Greedy algorithm also converge quickly, but their performance is poor. The proposed algorithm on average reduces the UEs’ overall energy consumption by 59.87%, 57.28%, and 82.89%, compared to Random, Greedy, and Local. In addition, since Local algorithm executes all computing tasks locally, the performance of it can be seen as an upper bound and is not affected by the number of iterations.

Figure 4 shows the trend of UEs’ overall energy consumption under different UAV’s storage capacity. Meanwhile, the storage capacity of the BS is also changing, which is twice that of the UAV. With the increase of storage capacity, UEs’ overall energy consumption by all algorithms except Local has been decreased. This is because, the UAV and BS can store more MEC services required by UEs with larger storage capacity. Compared with Random, Greedy and Local, UEs’ overall energy consumption of the proposed algorithm can be reduced by 60.10%, 60.16%, 78.74%, respectively. As shown in Figure 5, compared with the AGC-MEC architecture of one UAV, UEs’ overall energy consumption of the AGC-MEC architecture of two UAVs is reduced by 20.5% on average under different storage capacity schemes. However,
when the storage capacity of UAV increases to 10, the performance of one UAV and two UAVs is close. This is because when the storage capacity of a UAV increases to 10, it can store more services and enhance computing power. The original computing tasks may be completed by one UAV.

Figure 6 demonstrates the trend of UEs’ overall energy consumption under different UEs’ workload (i.e., the number of required CPU cycles). In this simulation, we use the workload increasing coefficient to represent the change of workload. As shown in Figure 6, with the increase of UEs’ workload, the UEs’ overall energy consumption increases. This is because the increase of workload may lead to that more UEs perform tasks locally. Compared with Random, Greedy and Local, the proposed algorithm on average reduces the UEs’ overall energy consumption by 48.01%, 44.17%, 65.78%, respectively. Furthermore, we alleviate the increase of UEs’ workload by increasing the number of MEC servers. As shown in Figure 7, compared with the AGC-MEC architecture of one UAV, UEs’ overall energy consumption of the AGC-MEC architecture of two UAVs under different UEs’ workload is reduced by 19.12% on average. Therefore, when the UEs’ workload is too large, it can be effectively alleviated by increasing the number of MEC servers.

In addition, we also consider an air-ground collaborative MEC network, which is composed of one UAV, one BS, one mobile vehicle, and one balloon to provide computing services collaboratively. Bal-
loon related parameters are set as follows: the altitude is 1000m, the maximum computation capacity is [8, 10] GHz, and the maximum number of UEs associated with balloon is [4, 6] [12, 39]. Mobile vehicle related parameters are set as follows: the maximum computation capacity is [6, 8] GHz, the maximum number of UEs associated with vehicle is [3, 5], and the trajectory of the vehicle is set to a fixed trajectory. As shown in Figure 8, after adding a balloon and a mobile vehicle, the UE’s overall energy consumption decreases by 68.69%, 79.27%, and 32.25%, compared with the other collaborative MEC network. It can be seen that multiple MEC paradigms collaboration can significantly improve computing performance and reduce user energy consumption. However, increasing the number of MEC servers will also bring economic pressure, which requires efficient deployment according to the actual UE’s task requirements.

Figure 9 demonstrates the trajectory of the UAV optimized by the proposed algorithm and the fixed trajectory of the vehicle, where the AGC-MEC architecture includes one UAV, one BS, one mobile vehicle, and one balloon. To be specific, the initial and final horizontal positions of the UAV are (0,0) and (200,200), respectively. As shown in figure, the proposed algorithm enables the UAV to fly close to UEs. This is because that the UAV can provide better computing service in the close proximity of UEs. The trajectory of the vehicle is a square with coordinates (100, 100) as its center and 150m as its side length. In fact, mobile-ground collaboration involves traffic roads conditions, which needs the real traffic data to optimize the trajectory of vehicle. This problem is complex, and we will consider it in future work.

VI. POTENTIAL RESEARCH DIRECTIONS OF AGC-MEC

In despite of its great potential, the study on the AGC-MEC architecture is still in its infancy, where many problems should be solved. In this section, we introduce three potential research directions that can help translate the visions of the AGC-MEC into reality.

6.1 Network Control

As previously mentioned, the AGC-MEC network involves different MEC paradigms belonging to different segments distributed in a wide area and thus needs to be well managed. As is known to all, software defined networking (SDN) separating the data plane and control plane introduces a unified control plane interface and global view of the whole network. SDN is able to provide centralized network control and flexible resource management of the collaboration among different MEC paradigms. Still, the SDN-based AGC-MEC is faced with many critical issues. For example, the placement of SDN controllers is a key to the SDN-based AGC-MEC, which should take many factors into consideration such as energy efficiency, load balancing, latency, and scalability, etc [40].

6.2 Security and Privacy

Since MEC servers in the AGC-MEC are located at the network edge and have frequent interactions, they lack effective backup and recovery measures of their data, which is prone to be attacked or misused by malicious users. Furthermore, compared with the cloud
computing data center in the core network. MEC can collect more high-value sensitive information of UEs, including location information, lifestyle, social relations, even health status, etc. Therefore, the security and privacy protection is critical to the AGC-MEC architecture. Fortunately, blockchain is a burgeoning distributed ledger technology, which has the potential to ensure the data and resource exchange among untrusted MEC nodes being safe in the AGC-MEC.

6.3 Intelligent Collaboration

For a long-lasting sophisticated mission, it may require multiple MEC servers to provide services together. To be specific, some MEC servers may act as controllers to determine offloading decisions. According to the decisions, some MEC servers may be used as relays to offload task to servers with strong computing power and rich resources. And then these MEC servers may cooperate to provide computing services. Therefore, it is valuable to study how many MEC servers are needed to complete task and how to intelligently collaborate.

VII. CONCLUSION

In this article, we have proposed the AGC-MEC architecture to explore the complementary integration of all potentially available MEC servers within air and ground by various collaborative ways to provide high-quality intelligent services for future 6G network. We have described the AGC-MEC architecture and three typical use cases. The challenging issues and their potential solutions have also been discussed. Furthermore, we have conducted a case study of collaborative service placement for AGC-MEC, and presented several potential research directions for future study.

ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China under Grant 62171465, 62072303, 62272223, U22A2031.

References

[1] M. Z. Chowdhury, M. Shahjalal, et al., “6g wireless communication systems: Applications, requirements, technologies, challenges, and research directions,” IEEE Open Journal of the Communications Society, 2020, vol. 1, pp. 957–975.
[2] H. Yang, A. Alphones, et al., “Artificial intelligence enabled intelligent 6g networks,” IEEE Network, 2020, vol. 34, no. 6, pp. 272–280.
[3] C. Wang, X. Yu, et al., “Qos-aware offloading based on communication-computation resource coordination for 6g edge intelligence,” China Communications, 2023, vol. 20, no. 3, pp. 236–251.
[4] Y. Mao, C. You, et al., “A survey on mobile edge computing: The communication perspective,” IEEE Communications Surveys Tutorials, 2017, vol. 19, no. 4, pp. 2322–2358.
[5] N. H. Mahmood, H. Alves, et al., “Six key features of machine type communication in 6g,” in 2020 2nd 6G Wireless Summit (6G SUMMIT). Levi, Finland: IEEE, 2020, pp. 1–5.
[6] W. You, C. Dong, et al., “Joint task scheduling, resource allocation, and uav trajectory under clustering for fanets,” China Communications, 2022, vol. 19, no. 1, pp. 104–118.
[7] Y. Dai, B. Lin, et al., “Uav-assisted data offloading for smart container in offshore maritime communications,” China Communications, 2022, vol. 19, no. 1, pp. 153–165.
[8] F. Zhou, R. Hu, et al., “Mobile edge computing in unmanned aerial vehicle networks,” IEEE Wireless Communications, 2020, vol. 27, no. 1, pp. 140–146.
[9] Y. Qu, H. Dai, et al., “Service provisioning for uav-enabled mobile edge computing,” IEEE Journal on Selected Areas in Communications, 2021, vol. 39, no. 11, pp. 3287–3305.
[10] N. Cheng, W. Xu, et al., “Air-ground integrated mobile edge networks: Architecture, challenges, and opportunities,” IEEE Communications Magazine, 2018, vol. 56, no. 8, pp. 26–32.
[11] W. You, C. Dong, et al., “Joint optimization of area coverage and mobile-edge computing with clustering for fanets,” IEEE Internet of Things Journal, 2021, vol. 8, no. 2, pp. 695–707.
[12] S. Wang, M. Chen, et al., “Federated learning for task and resource allocation in wireless high altitude balloon networks,” IEEE Internet of Things Journal, 2021, vol. 8, no. 24, pp. 17460–17475.
[13] L. Zhao, K. Yang, et al., “Vehicular computation offloading for industrial mobile edge computing,” IEEE Transactions on Industrial Informatics, 2021, vol. 17, no. 11, pp. 7871–7881.
[14] H. Peng and X. Shen, “Multi-agent reinforcement learning based resource management in mec- and uav-assisted vehicular networks,” IEEE Journal on Selected Areas in Communications, 2021, vol. 39, no. 1, pp. 131–141.
[15] Z. Zhou, J. Feng, et al., “An air-ground integration approach for mobile edge computing in iot,” IEEE Communications Magazine, 2018, vol. 56, no. 8, pp. 40–47.
[16] F. Jiang, K. Wang, et al., “Deep-learning-based joint resource scheduling algorithms for hybrid mec networks,” IEEE Internet of Things Journal, 2020, vol. 7, no. 7, pp. 6252–6265.
[17] F. Jiang, K. Wang, et al., “Ai driven heterogeneous mec system with uav assistance for dynamic environment: Challenges and solutions,” IEEE Network, 2021, vol. 35, no. 1, pp. 400–408.
[18] M. Chen and Y. Hao, “Task offloading for mobile edge com-
puting in software defined ultra-dense network,” IEEE Journal on Selected Areas in Communications, 2018, vol. 36, no. 3, pp. 587–597.

[19] T. X. Tran and D. Pompili, “Joint task offloading and resource allocation for multi-server mobile-edge computing networks,” IEEE Transactions on Vehicular Technology, 2019, vol. 68, no. 1, pp. 856–868.

[20] J. Zhang and K. B. Letaief, “Mobile edge intelligence and computing for the internet of vehicles,” Proceedings of the IEEE, 2020, vol. 108, no. 2, pp. 246–261.

[21] R. Kianfar, B. Augusto, et al., “Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge,” IEEE Transactions on Intelligent Transportation Systems, 2012, vol. 13, no. 3, pp. 994–1007.

[22] B. Shang and L. Liu, “Mobile-edge computing in the sky: Energy optimization for air–ground integrated networks,” IEEE Internet of Things Journal, 2020, vol. 7, no. 8, pp. 7443–7456.

[23] B. Yang, C. A. Cao, et al., “Offloading optimization in edge computing for deep-learning-enabled target tracking by internet of uavs,” IEEE Internet of Things Journal, 2021, vol. 8, no. 12, pp. 9878–9893.

[24] X. Hu, K. Wong, et al., “Uav-assisted relaying and edge computing: Scheduling and trajectory optimization,” IEEE Transactions on Wireless Communications, 2019, vol. 18, no. 10, pp. 4738–4752.

[25] L. Zhao, K. Yang, et al., “A novel cost optimization strategy for sdn-enabled uav-assisted vehicular computation offloading,” IEEE Transactions on Intelligent Transportation Systems, 2021, vol. 22, no. 6, pp. 3664–3674.

[26] X. Hu, K. Wong, et al., “Wireless-powered edge computing with cooperative uav: Task, time scheduling and trajectory design,” IEEE Transactions on Wireless Communications, 2020, vol. 19, no. 12, pp. 8083–8098.

[27] X. Hu, K. Wong, et al., “Wireless-powered mobile edge computing with cooperated uav,” in 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2019, pp. 1–5.

[28] Y. Liu, S. Xie, et al., “Cooperative offloading and resource management for uav-enabled mobile edge computing in power iot system,” IEEE Transactions on Vehicular Technology, 2020, vol. 69, no. 10, pp. 12229–12239.

[29] H. Dai, J. Yu, et al., “Bloom filter with noisy coding framework for multi-set membership testing,” IEEE Transactions on Knowledge and Data Engineering, 2023, vol. 35, no. 7, pp. 6710–6724.

[30] G. Li, G. Xu, et al., “Edgelaas: Edge learning as a service for knowledge-centric connected healthcare,” IEEE Network, 2019, vol. 33, no. 6, pp. 37–43.

[31] J. Wang, K. Liu, et al., “Online uav-mounted edge server dispatching for mobile-to-mobile edge computing,” IEEE Internet of Things Journal, 2020, vol. 7, no. 2, pp. 1375–1386.

[32] S. Deng, H. Zhao, et al., “Edge intelligence: The confluence of edge computing and artificial intelligence,” IEEE Internet of Things Journal, 2020, vol. 7, no. 8, pp. 7457–7469.

[33] C. Dong, Y. Shen, et al., “Uavs as an intelligent service: Boosting edge intelligence for air-ground integrated networks,” IEEE Network, 2021, vol. 35, no. 4, pp. 167–175.

[34] C. Xiang, Y. Zhou, et al., “Reusing delivery drones for urban crowdsensing,” IEEE Transactions on Mobile Computing, 2023, vol. 22, no. 5, pp. 2972–2988.

[35] Z. Yang, C. Pan, et al., “Energy efficient resource allocation in uav-enabled mobile edge computing networks,” IEEE Transactions on Wireless Communications, 2019, vol. 18, no. 9, pp. 4576–4589.

[36] L. Wang, K. Wang, et al., “Deep reinforcement learning based dynamic trajectory control for uav-assisted mobile edge computing,” IEEE Transactions on Mobile Computing, 2022, vol. 21, no. 10, pp. 3536–3550.

[37] J. Hu, M. Jiang, et al., “Joint optimization of uav position, time slot allocation, and computation task partition in multiuser aerial mobile-edge computing systems,” IEEE Transactions on Vehicular Technology, 2019, vol. 68, no. 7, pp. 7231–7235.

[38] V. Farhadi, F. Mehmeti, et al., “Service placement and request scheduling for data-intensive applications in edge clouds,” in IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, 2019, pp. 1279–1287.

[39] Y. Zhou, X. Huang, et al., “5g-based measurements and characterizations of low-altitude tethered balloon multipath channel,” in 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), 2021, pp. 1–5.

[40] T. Das, V. Sridharan, et al., “A survey on controller placement in SDN,” IEEE Communications Surveys Tutorials, 2020, vol. 22, no. 1, pp. 472–503.

Biographies

Qin Zhen received the M.S. and Ph.D degrees in communication engineering from Army Engineering University, Nanjing, China, in 2019 and 2022, respectively. She is currently a lecturer with the Department of Information and Communication, Noncommissioned Officer Academy of PAP, Hangzhou. Her research interests include unmanned aerial vehicle communications, mobile edge computing, and air-ground integrated networks.

He Shoushuai received the B.S. degree in information security from Harbin Institute of Technology, in 2017. He is currently pursuing the Ph.D degree with Army Engineering University of PLA. His research interests include electromagnetic spectrum cognition and deep learning.
**Wang Hai** received the Ph.D degree from Nanjing Institute of Communications Engineering, China, in 1999. He is now the professor and doctoral supervisor of Army Engineering University of PLA. His research interests include wireless networking, ad hoc networking and vehicular communications technologies.

**Qu Yuben** received the B.S. degree in Mathematics and Applied Mathematics from Nanjing University, and both the M.S. degree in Communication and Information Systems and the Ph.D degree in Computer Science and Technology from Nanjing Institute of Communications, in 2009, 2012 and 2016, respectively. During June 2019 to June 2022, he was a Postdoctoral Fellow with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. He is currently an associate research fellow in the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, and also with the Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space, Ministry of Industry and Information Technology, China. From October 2015 to January 2016, he was a visiting research associate in the School of Computer Science and Engineering, the University of Aizu, Japan. He was a recipient of the Best Paper Awards of GPC 2020, IEEE SAGC 2021, and IEEE/CIC ICCC 2023. His research interests mainly include mobile edge computing, edge intelligence, and UAV’s collaborative intelligence.

**Dai Haipeng** received the B.S. degree in the Department of Electronic Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2010, and the Ph.D degree in the Department of Computer Science and Technology in Nanjing University, Nanjing, China, in 2014. He is an associate professor in the Department of Computer Science and Technology in Nanjing University. He is an IEEE senior member and an ACM member. He received Best Paper Award from IEEE ICNP’15, Best Paper Award Runner-up from IEEE SECON’18, and Best Paper Award Candidate from IEEE INFOCOM’17.

**Xiong Fei** received the B.S. and M.S. degrees in computer science and technology from PLA University of Science and Technology, Nanjing, China in 2006 and 2009, respectively. He received the Ph.D degree in science of military command from the Army Engineering University of PLA, Nanjing, China in 2020. His research interests include wireless networking and vehicular communications technologies.

**Wei Zhenhua** received both the B.S. degree and the M.S. degree in Electronic Engineering from National University of Defense Technology in 2002 and 2006, respectively. He is currently an associate professor in the Xi’an Research Institute of High Technology, China. His research interests include anti-jamming and wireless communication.

**Li Hailong** received Ph.D degree in Computer Science and Technology from Xi’an Research Institute of High Technology in 2014. He is currently an associate professor in the Xi’an Research Institute of High Technology, China. His research interests include complex network and network security.