Priority-based Load Balancing with Multi-Agent Deep Reinforcement Learning for Space-Air-Ground Integrated Network Slicing

Haiyan Tu, Paolo Bellavista, *Senior Member, IEEE*, Liqiang Zhao, *Member, IEEE*, Gan Zheng, *Fellow, IEEE*, Kai Liang, *Member, IEEE*, Kai-Kit Wong, *Fellow, IEEE*

***Abstract*—Space-air-ground integrated network (SAGIN) slicing has been studied for supporting diverse applications, which consists of the terrestrial layer (TL) deployed with base stations (BS), the aerial layer (AL) deployed with unmanned aerial vehicles (UAV), as well as the space layer (SL) deployed with low earth orbit (LEO) satellites. The capacity of each SAGIN component is limited, and efficient and synergic load balancing has not been fully considered yet in the exiting literature. For this motivation, we originally propose a priority-based load balancing scheme for SAGIN slicing, where the AL and SL are merged into one layer, namely non-TL (NTL). Firstly, three typical slices (i.e., high-throughput, low-delay, and wide-coverage slices) are built under the same physical SAGIN. Then, a priority-based cross-layer load balancing approach is introduced, where the users will have the priority to access the terrestrial BS, and different slices have different priorities. More specifically, the overloaded BS can offload the users of low-priority slices to the NTL. Furthermore, the throughput, delay, and coverage of the corresponding slices are jointly optimized by formulating a multi-objective optimization problem (MOOP). In addition, due to the independence and priority relationship of TL and NTL, the above MOOP is decoupled into two sub-MOOPs. Finally, we customize a two-layer multi-agent deep deterministic policy gradient (MADDPG) algorithm for solving the two sub-problems, which firstly optimizes the user-BS association and resource allocation at the TL, then it determines the UAVs' position deployment, users-UAV/LEO satellite association, and resource allocation at the NTL. The reported simulation results show the advantages of our proposed LB scheme and show that our proposed algorithm outperforms the benchmarkers.**

***Index Terms*—Space-air-ground integrated networks, radio access network slicing, load balancing, multi-objective optimization, multi-agent deep deterministic policy gradient.**

I. Introduction

With the explosive growth of user equipment and service types in next generation wireless communications, the existing terrestrial networks will face great challenges [1]. Recently, the concept of space-air-ground integrated networks (SAGIN) [2], [3] was proposed for supporting growing service demand and ubiquitous coverage. Considering the location of network components, SAGIN is roughly divided into three layers: a terrestrial layer (TL), an aerial layer (AL), and a space layer (SL), which consist of multiple base stations (BSs), unmanned aerial vehicles (UAVs), and low earth orbit (LEO) satellites, respectively. With the integration of the three-layer architecture, SAGIN is capable to provide seamless coverage and enhance data transmission. In the following, we will consider a two-layer SAGIN system: the TL and non-TL (NTL), where NTL integrates the AL and SL.

As an enabling technology for next-generation wireless networks, radio access network (RAN) slicing [4] can support diversity services by constructing several independent logical sub-networks under the same physical infrastructure. As such, RAN slicing has been applied to support diverse SAGIN services under constraints of resource limitations [5], [6]. However, a realistic limitation is ignored in the existing works for SAGIN slicing, which is the capacity of each network component (BSs, UAVs, LEO satellites) is limited [7]. Due to the load imbalance, lightly loaded cells associated with a small number of users waste the remaining resources [8], while the users associated with heavily loaded cells compete for insufficient network resources, resulting in performance degradation [9]. Thus, effective and synergic load balancing (LB) schemes, capable of considering holistically all the SAGIN layers, are demanded to avoid overloading cells and the depletion of wireless resources. A few works have studied SAGIN LB, such as [10], but almost of them were concentrated on LEO satellite network, i.e., intra-SL LB solution. Inspired by the above considerations, we originally propose a priority-based cross-layer LB scheme for SAGIN slicing, which fully considers the disadvantages of the NTL [11].

Moreover, the types of applications/services keep expanding with the emerging networks, whose objectives may tend to conflict with each other. Multi-objective optimization problems (MOOPs) have been widely used to jointly optimizing multiple metrics [12]. The MOOPs have been addressed with scalar-oriented solutions by transforming to single-objective optimization problems (SOOPs) [13], [14]. However, the scalarization functions are designed with prior knowledge and the associated solution is highly dependent on the selected weights. Then, multi-agent deep reinforcement learning (MADRL) algorithms [15], [16] recently are popular for solving such

Haiyan Tu, Liqiang Zhao and Kai Liang are with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China (e-mail: hytu@stu.xidian.edu.cn; lqzhao@mail.xidian.edu.cn; kliang@mail.xidian.edu.cn.)

Paolo Bellavista is with the Department of Computer Science and Engineering, University of Bologna, Bologna 40126, Italy (e-mail: paolo.bellavista@unibo.it.)

Gan Zheng is with the School of Engineering, University of Warwick, Coventry, CV4 7AL, UK (e-mail: gan.zheng@warwick.ac.uk.)

Kai-Kit Wong is affiliated with the Department of Electronic and Electrical Engineering, University College London, Torrington Place, WC1E 7JE, United Kingdom and he is also affiliated with Yonsei Frontier Lab, Yonsei University, Seoul, Korea (e-mail: kai-kit.wong@ucl.ac.uk.)

problems.

In this article, we present a priority-based cross-layer LB scheme for SAGIN slicing. We construct the high-throughput, low-delay, and wide-coverage slice in the system, by sharing the same underlying two-layer physical SAGIN. Then, a priority-based cross-layer LB manner is established. In addition, we formulate a MOOP to optimize the throughput, the average delay, and the coverage, where the user-network components association, UAVs' positions deployment, subcarriers and power allocation are jointly optimized. In order to solve the MOOP, we propose a novel two-layer multi-agent deep deterministic policy gradient (MADDPG) algorithm. With the above background, the main contributions of this article are summarized as follows:

1) ***Two-layer SAGIN-based RAN slicing framework***: We propose a two-layer SAGIN slicing scheme, where the two layers are referred to TL and NTL (the AL and SL are combined as the NTL). Specifically, three slices, namely, the high-throughput slice, low-delay slice, and wide-coverage slice, are constructed to provide corresponding services relying on the two-layer physical SAGIN.
2) ***Priority-based cross-layer load balancing***: We consider the LB between TL and NTL, taking into account the priority of user-TL associations and the priorities of slices. To elaborate, users have the highest priority to access to BSs of the TL by considering the disadvantages of the NTL. Then, the overloaded BSs can offload the excess users to the NTL. Beyond that, we set different priorities for slices, and those with low priorities will be offloaded to the NTL preferentially.
3) ***MOOP formulation and decomposition***: We establish the MOOP to jointly optimize the throughput, average delay, and coverage percentage for corresponding slices by dynamically considering the user-BS/UAV/LEO satellite association, as well as the UAVs' positions deployment, subcarriers and power resources allocation. Furthermore, due to the relationship of the TL and NTL, the above problem is decomposed into two sub-MOOPs.
4) ***Two-layer MADDPG framework for sub-problems***: We customize a two-layer MADDPG framework to handle the associated sub-problems. The algorithm first determines the user-BS association and resource allocation for the users on each slice at the TL, where the proposed LB scheme is applied; then, it proceeds with the UAVs' positions deployment, user-UAV/LEO satellite association, and resource allocation for the offloaded and unconnected users at NTL. With the alternating optimization of the two layers, we will show that the above problems can be adequately solved.

The remainder of this article is organized as follows. In the next section, we briefly introduce the related work. In Section III, the two-layer SAGIN slicing model is described, and three different RAN slices are shown in detail. In Section IV, we formulate the MOOP for SAGIN slicing. In Section V, we decompose the MOOP into two sub-problems and solve by the proposed two-layer MADDPG algorithm. Section VI with performance evaluation and Section VII concludes this article.

II. Related Works

In this section, we will illustrate the state-of-the-art in SAGIN slicing/LB, as well as the introduction of the most popular methods for solving the MOOPs in SAGIN.

A. SAGIN slicing

RAN slicing can be applied to support diverse customized services in SAGIN under various resource constraints [17], [18]. Zhang *et al.* [19] studied the co-existence of enhanced mobile broadband (eMBB) and ultra reliable and low-latency (URLLC) slices in SAGIN, and proposed a heterogeneous traffic offloading approach by efficiently optimizing the resource allocation and the UAVs trajectory design. Cao *et al.* [20], [21] proposed a novel resource allocation and orchestration framework, named as Slice-Soft-SAGIN, to provide reliable and efficient resource allocation and orchestration for the requested slices. Zhou *et al.* [5] considered three classes of RAN slices in SAGIN, and proposed a joint central and distributed MADDPG algorithm to find the Pareto optimal solutions. In [22], the authors focused on the URLLC slice in multi-access edge computing (MEC) network operating at the SAGIN environment, and a novel distributed dynamic network slicing algorithm was proposed to maximize network payoff.

B. SAGIN load balancing

The capacity of each network component in SAGIN are limited, thus an efficient LB scheme should be designed for the overloaded cells. As in [23], Sun *et al.* constructed a handover model for latency-sensitive services in a large-scale LEO satellite network (LEO-SN), taking into account the load balancing indicator and the Carrier-to-Noise Ratio of LEO satellite links. Wang *et al.* [24] investigated edge-computing load balancing problem for LEO-SN, and the Ford-Fulkerson algorithm was used to obtain the transmission and computing resource allocation strategy. Focusing on the issues of high-speed mobility, topology dynamic change and link overload problems in LEO-SN, an intelligent decentralized load balancing routing scheme was designed with deep reinforcement learning in [25]. In addition, Tao *et al.* [26] designed a load balancing based traffic scheduling method for software defined networking (SDN)-based SAGIN architecture to deploy inter-satellite links between LEO satellites. The above works focused on the load balancing of the LEO-SN (i.e., intra-SL LB). Though, Zuo *et al.* [27] presented a novel cross-layer traffic offloading approach in SAGIN, which has not considered the priority settings completely. Encouraged by these, we develop a priority-based cross-layer LB scheme for SAGIN slicing, where users are having the priority to access the TL and different slices are set with different priority level for offloading.

C. MOOP in SAGIN

As each SAGIN slice has diverse performance requirement, MOOPs are eminently suitable for the scenario. The available scalar methods for solving the MOOPs in SAGIN have been well studied. Tang *et al.* [28] simultaneously analyzed the end-to-end (E2E) delay and transmission rate in SAGIN, but only the E2E delay was employed as the objective, while the transmission rate was treated as a constraint. Zhou *et al.* [29] have also used the constrained optimization method for minimizing the time-averaged delay subject to the constraint of a time-averaged energy consumption in SAGIN. Whilst, the MADDPG algorithm was employed in *et al.* [30] to maximize the number of tasks with the delay constraints for each UAV in SAGIN for Internet of Remote Things. Paul *et al.* [31] optimized the total latency for the DT-assisted SAGIN by a MADRL algorithm, although both the network coverage and energy consumption were considered.

Actually, the non-scalar intelligent algorithms, such as MADRL algorithms, are capable to simultaneously optimize different optimization objectives for the SAGIN [5]. Thus, we design the novel MADDPG algorithm with a two-layer structure for the proposed two-layer SAGIN slicing architecture.

III. System Model

We consider the downlink (DL) SAGIN slicing model as shown in Fig. 1, which consists of a two-layer RAN, including the TL of M BSs, the non-TL with V UAVs and O LEO satellites. Importantly, the cross-layer load balancing is considered in this model, i.e., the overloaded BSs will offload part of loads to the UAVs or LEO satellites. The set of M, V and O nodes are denoted as $\mathbf{M} = \{\mathbf{1}, \mathbf{2}, \cdots, \mathbf{M}\}$, $\mathbf{V} = \{\mathbf{1}, \mathbf{2}, \cdots, \mathbf{V}\}$, and $\mathbf{O} = \{\mathbf{1}, \mathbf{2}, \cdots, \mathbf{O}\}$, respectively. It is assumed that different types of nodes operate at different frequency bands with available bandwidth B Hz to avoid the cross-layer interference, which is divided into N orthogonal subcarriers with bandwidth $B_N = B/N$ Hz. The total power of each BS, UAV and LEO satellite is P_B, P_V and P_O.

In this system, the mobile network operator (MNO) provides several slices for the different services, which are mainly mapped to three slices. In detail, the high-throughput, low-delay, and wide-coverage slices are created, which are referred as slice s, $s \in \{H, L, C\}$. We assume that K terrestrial users with set of $\mathbf{K}$ are randomly distributed in the area of S_p and can move between time slots at a certain speed, but they are stationary within a time slot. We denote $K_s(t)$ as the number of users requesting the slice s at time slot t and the set can be defined as $\mathbf{K_s}(t) = \{1, 2, \cdots, K_s(t)\}$.

Normally, the positions of BSs are fixed, and the position of UAVs will be one of decision variables, which are optimized to maximize the objectives. Consequently, the positions of the BSs, users can be modeled by 2-dimensional coordinates as: $\mathbf{\Lambda_{BS}} = \left\{ \left[x_m^{BS}, y_m^{BS}\right], \forall m \in \mathbf{M} \right\}$ and $\mathbf{\Lambda_{UE}}(t) = \left\{ \left[x_{k,s}^{UE}(t), y_{k,s}^{UE}(t)\right], \forall k \in \mathbf{K}, s \in \{H, L, C\} \right\}$. Meanwhile, the positions of UAVs and LEO satellites can be represented by 3-dimensional coordinates as: $\mathbf{\Lambda_{UAV}}(t) = \left\{ \left[x_v^{UAV}(t), y_v^{UAV}(t), z_v^{UAV}(t)\right], \forall v \in \mathbf{V} \right\}$, and $\mathbf{\Lambda_{LEO}} = \left\{ \left[x_o^{LEO}, y_o^{LEO}, z_o^{LEO}\right], \forall o \in \mathbf{O} \right\}$, respectively. To simplify the problem, we does not consider the communication between the network components (such as the BS-BS and BS-UAV) [32], and the motion trajectories of the users and UAVs will not be analyzed.

A. User-BS communication

For the user-BS DL communication, the subcarriers are modeled as classic Rayleigh fading channel [33], and the channel fading $h_{k,s,m,n}^{BS}(t)$ for user k on slice s with BS m and subcarrier n at time slot t follows an exponential distribution with unity mean. Then, the channel coefficient $g_{k,s,m,n}^{BS}(t)$ can be calculated as:

$$g_{k,s,m,n}^{BS}(t) = h_{k,s,m,n}^{BS}(t) \left[d_{k,s,m}^{BS}(t)\right]^{-\alpha}, \tag{1}$$

where $d_{k,s,m}^{BS}(t) = \sqrt{\left(x_m^{BS} - x_{k,s}^{UE}(t)\right)^2 + \left(y_m^{BS} - y_{k,s}^{UE}(t)\right)^2}$ is the distance between BS m and user k on slice s, and α is the path loss exponent.

Then, we denote $p_{k,s,m,n}^{BS}(t)$ as the transmit power of the BS m allocate to user k on slice s with subcarrier n at time slot t, and the set can be defined as $\mathbf{P_s^{BS}}(t) = \{p_{k,s,m,n}^{BS}(t)\}$. The signal to interference plus noise power ratio (SINR) and data rate of user k associated with BS m and subcarrier n on slice s at time slot t are

$$\gamma_{k,s,m,n}^{BS}(t) = \frac{p_{k,s,m,n}^{BS}(t) g_{k,s,m,n}^{BS}(t)}{I_{k,s,m,n}^{BS}(t) + (\frac{B}{N})N_0}, \tag{2}$$

and

$$r_{k,s,m,n}^{BS}(t) = \frac{B}{N} \log_2 \left[1 + \frac{p_{k,s,m,n}^{BS}(t) g_{k,s,m,n}^{BS}(t)}{I_{k,s,m,n}^{BS}(t) + (\frac{B}{N})N_0}\right], \tag{3}$$

where $I_{k,s,m,n}^{BS}(t)$ is the interference of user k on slice s at time slot t, and N_0 is the power spectral density of the additive white Gaussian noise (AWGN).

B. User-UAV communication

For the user-UAV DL communication, the fading of $h_{k,s,v,n}^{UAV}(t)$ for user k on slice s associated with UAV v and subcarrier n at time slot t follows a Rician channel model. Accordingly, the channel coefficient $g_{k,s,v,n}^{UAV}(t)$ can be calculated as:

$$\begin{aligned} g_{k,s,v,n}^{UAV}(t) &= h_{k,s,v,n}^{UAV}(t) \left[d_{k,s,v}^{UAV}(t)\right]^{-\alpha} \\ &= h_0 \left[d_{k,s,v}^{UAV}(t)\right]^{-\alpha} \left(\frac{R}{R+1} \hat{h}_{k,s,v,n}^{UAV}(t) + \frac{1}{R+1} \tilde{h}_{k,s,v,n}^{UAV}(t)\right), \end{aligned} \tag{4}$$

where h_0 represents the reference channel gain with the distance of 1 meter; R denotes the Rician fading factor; $\hat{h}_{k,s,v,n}^{UAV}(t)$ denotes the line-of sight (LoS) factor with $|\hat{h}_{k,s,v,n}^{UAV}(t)| = 1$; and $\tilde{h}_{k,s,v,n}^{UAV}(t) \sim \mathcal{CN}(0,1)$ indicates the non-line-of-sight (NLoS) component. Moreover, $d_{k,s,v}^{UAV}(t)$ is the distance between UAV v to user k, and the expression can be

$$\begin{aligned} d_{k,s,v}^{UAV}(t) = \Big[&\left(x_v^{UAV}(t) - x_{k,s}^{UE}(t)\right)^2 \\ &+ \left(y_v^{UAV}(t) - y_{k,s}^{UE}(t)\right)^2 + \left(z_v^{UAV}(t)\right)^2 \Big]^{1/2}. \end{aligned} \tag{5}$$

Further, we define the transmit power of the UAV v allocate to user k on slice s with subcarrier n at time slot t by $p_{k,s,m,n}^{UAV}(t)$, with the set of $\mathbf{P_s^{UAV}}(t) = \{p_{k,s,m,n}^{UAV}(t)\}$. Then, the SINR and data rate of user k associated with UAV v and subcarrier n on slice s at time slot t can be calculated as:

$$\gamma_{k,s,v,n}^{UAV}(t)=\frac{p_{k,s,v,n}^{UAV}(t)g_{k,s,v,n}^{UAV}(t)}{I_{k,s,v,n}^{UAV}(t)+(\frac{B}{N})N_0}, \tag{6}$$

and

$$r_{k,s,v,n}^{UAV}(t)=\frac{B}{N}\log_2\left[1+\frac{p_{k,s,v,n}^{UAV}(t)g_{k,s,v,n}^{UAV}(t)}{I_{k,s,v,n}^{UAV}(t)+(\frac{B}{N})N_0}\right], \tag{7}$$

where $I_{k,s,v,n}^{UAV}(t)$ is the interference of user k on slice s associated with UAV v and subcarrier n at time slot t.

C. User-LEO communication

As for the user-LEO DL communication, the fading of $h_{k,s,o,n}^{LEO}(t)$ for user k on slice s associated with LEO satellite o and subcarrier n at time slot t is the unit radio propagation loss of the satellite link, which suffers free space loss. Then, the channel coefficient $g_{k,s,o,n}^{LEO}(t)$ can be calculated as:

$$g_{k,s,o,n}^{LEO}(t)=h_{k,s,o,n}^{LEO}(t)\left[d_{k,s,o}^{LEO}(t)\right]^{-\alpha}=\left(\frac{c}{4\pi f_c}\right)^2\left[d_{k,s,o}^{LEO}(t)\right]^{-\alpha}, \tag{8}$$

where c is the velocity of light and f_c is the carrier frequency. Moreover, due to the high altitude of the LEO satellites, the distance from LEO satellite o to user k on slice s is approximated as $d_{k,s,o}^{LEO}(t)\approx|z_o^{LEO}(t)|$.

Next, $p_{k,s,o,n}^{LEO}(t)\in\mathbf{P_s^{LEO}}(t)$ is the power allocation of the LEO satellite o to user k on slice s with subcarrier n at time slot t. Subsequently, the SINR and data rate of user k associated with LEO satellite o and subcarrier n on slice s at time slot t can be formulated as:

$$\gamma_{k,s,o,n}^{LEO}(t)=\frac{p_{k,s,o,n}^{LEO}(t)g_{k,s,o,n}^{LEO}(t)}{I_{k,s,o,n}^{LEO}(t)+(\frac{B}{N})N_0}, \tag{9}$$

and

$$r_{k,s,o,n}^{LEO}(t)=\frac{B}{N}\log_2\left[1+\frac{p_{k,s,o,n}^{LEO}(t)g_{k,s,o,n}^{LEO}(t)}{I_{k,s,o,n}^{LEO}(t)+(\frac{B}{N})N_0}\right], \tag{10}$$

where $I_{k,s,o,n}^{LEO}(t)$ is the interference of user k on slice s associated with LEO satellite o and subcarrier n at time slot t.

D. User association with priority-based LB scheme

As we do not consider the association between the network components, so only the association between users and network components needs to be analyzed. In our priority-based LB scheme, two kinds of priorities are considered: association priority and offloading priority. Firstly, the user-BS association are set the highest priority, i.e., users have priority to access the TL, the reason is that the BSs are already deployed, and the links of UAVs are unstable, as well as the long distance between LEO satellites and users. As for the offloading priority, we set each slice with different offloading priorities, wherein the users on low-priority slices will be offloaded first. If the BSs are overloaded, the offloading procedure will be triggered. We set the slice H with the highest priority in the system, followed by slice L, and, slice C has the lowest priority. Importantly, we assume that

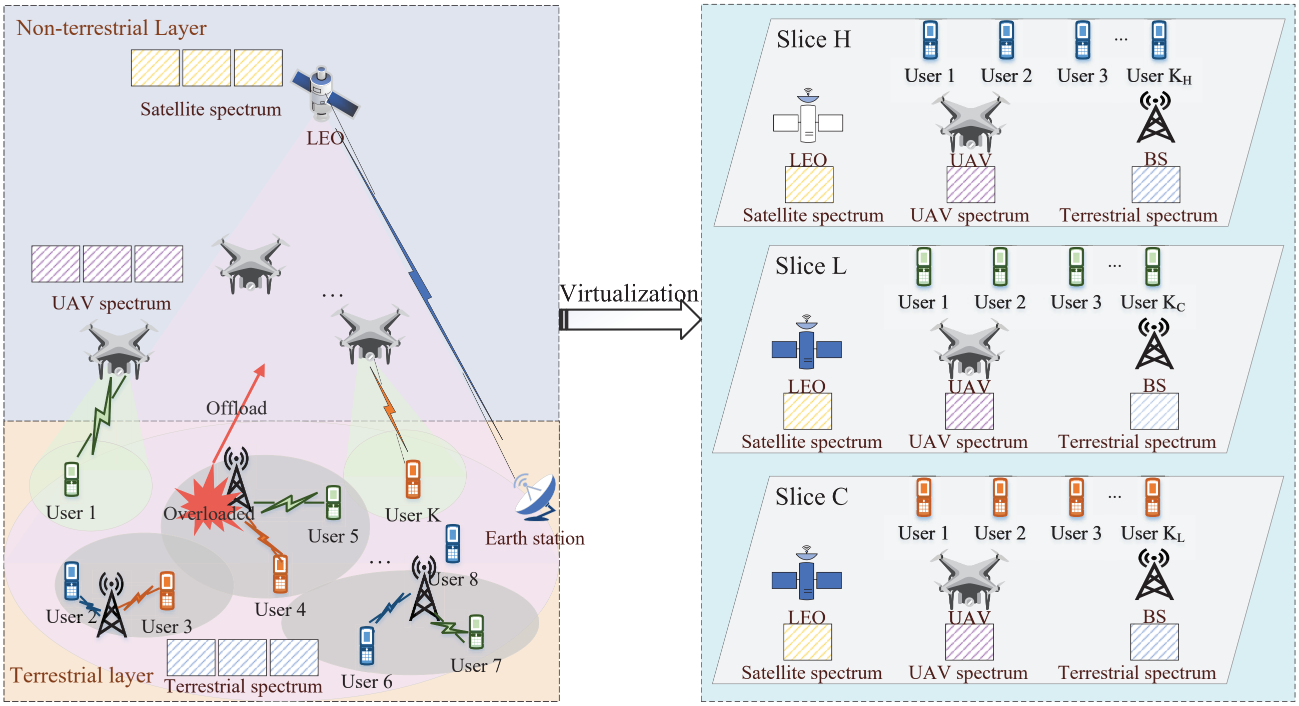

Fig. 1. System model for the SAGIN slicing.

each user can only access at most one network component. Then, the set of association indicators can be defined as $\mathbf{\Psi_s}(t)$=$\{\varphi^{j}_{k,s,i}(t), \forall(i,j) \in \{(m,\mathrm{BS}),(v,\mathrm{UAV}),(o,\mathrm{LEO})\}\}$. Furthermore, we denote a binary factor $\xi_{k,s,n}(t)$ as subcarrier allocation indicators of user k on slice s allocated with subcarrier n at time slot t, and the set is represented as $\mathbf{\Xi_s}(t)$=$\{\xi_{k,s,n}(t)\}$. The specific expression can be written as:

$$\varphi^{j}_{k,s,i}(t)=\begin{cases}1, \text{ if user } k \text{ on slice } s \text{ is associated with node } i, \\ \quad \text{for}\,(i,j) \in \{(m,\mathrm{BS}),(v,\mathrm{UAV}),(o,\mathrm{LEO})\}\,, \\ 0, \text{ otherwise,}\end{cases} \tag{11}$$

and

$$\xi_{k,s,n}(t)=\begin{cases}1, \text{if subcarrier } n \text{ is allocated to user } k \text{ on slice } s, \\ 0, \text{otherwise}\end{cases} \tag{12}$$

Therefore, the SINR of user k on slice s associated with BSs, UAVs and LEO satellites at time slot t are calculated as, respectively:

$$\gamma^{BS}_{k,s}(t) = \sum_{n\in\mathbf{N}}\sum_{m\in\mathbf{M}} \varphi^{BS}_{k,s,m}(t)\xi_{k,s,n}(t)\gamma^{BS}_{k,s,m,n}(t), \tag{13}$$

$$\gamma^{UAV}_{k,s}(t) = \sum_{n\in\mathbf{N}}\sum_{v\in\mathbf{V}} \varphi^{UAV}_{k,s,v}(t)\xi_{k,s,n}(t)\gamma^{UAV}_{k,s,v,n}(t), \tag{14}$$

$$\gamma^{LEO}_{k,s}(t) = \sum_{n\in\mathbf{N}}\sum_{o\in\mathbf{O}} \varphi^{LEO}_{k,s,o}(t)\xi_{k,s,n}(t)\gamma^{LEO}_{k,s,o,n}(t). \tag{15}$$

Similarly, the downlink data rate of user k on slice s associated with BSs, UAVs and LEO satellites at time slot t are defined respectively as

$$R^{BS}_{k,s}\left(t\right) = \sum_{n\in\mathbf{N}}\sum_{m\in\mathbf{M}} \varphi^{BS}_{k,s,m}(t)\xi_{k,s,n}(t)r^{BS}_{k,s,m,n}(t), \tag{16}$$

$$R^{UAV}_{k,s}\left(t\right) = \sum_{n\in\mathbf{N}}\sum_{v\in\mathbf{V}} \varphi^{UAV}_{k,s,v}(t)\xi_{k,s,n}(t)r^{UAV}_{k,s,v,n}(t), \tag{17}$$

$$R^{LEO}_{k,s}\left(t\right) = \sum_{n\in\mathbf{N}}\sum_{o\in\mathbf{O}} \varphi^{LEO}_{k,s,o}(t)\xi_{k,s,n}(t)r^{LEO}_{k,s,o,n}(t). \tag{18}$$

IV. Problem Formulation and Performance Analysis

Since the key performance indicators for each slice are greatly different, the primary optimization objectives for each slice are obviously different as well. In this section, we firstly introduce the key performance indicators for each slice, and then the multi-objective optimization problem is formulated.

A. High-throughput slice

For the high-throughput slice H, the users require to transmit high-throughput data through the allocated subcarriers. Hence, the key optimization objective for slice H is the throughput.

According to (16), (17), and (18), we can derive the total data rate of user k on slice H as

$$R_{k,H}(t) = R^{BS}_{k,H}(t) + R^{UAV}_{k,H}(t) + R^{LEO}_{k,H}(t). \tag{19}$$

Then, the throughput of slice H can be expressed as

$$R^{sum}_{H}\left(t\right) = \sum_{k\in\mathbf{K_H}} R_{k,H}\left(t\right). \tag{20}$$

B. Low-delay slice

For the low-delay slice L, the users hope to suffer relatively low service delay. Firstly, the data rate of user k on slice L is $R_{k,L}(t) = R^{BS}_{k,L}(t) + R^{UAV}_{k,L}(t) + R^{LEO}_{k,L}(t)$. Next, we assume $\mathbf{A_L(t)} = \{A_{1,L}(t), A_{2,L}(t), \cdots, A_{K_L,L}(t)\}$ as the process of random data arrivals on slices L, where $A_{k,L}(t)$ follows a Poisson arrival process with rate of $\lambda_{k,L}$ and is assumed to be independent among the users.

In addition, the service procedure can be modeled as an M/D/1 queue [34]. The service delay $D_{k,L}(t)$ of user k on slices L at time slot t is denoted as:

$$D_{k,L}\left(t\right) = \frac{d_{k,L}\left(t\right)}{c} + \frac{A_{k,L}\left(t\right)}{R_{k,L}\left(t\right)} + \frac{\lambda_{k,L}\left(t\right)}{2\left[\left(R_{k,L}\left(t\right)\right)^2 - \lambda_2 R_{k,L}\left(t\right)\right]}, \tag{21}$$

where the first item of (21) is the propagation delay. The second item is the transmission delay and the last item is the queuing delay. Moreover, $d_{k,L}\left(t\right)$ is the distance between user k and the associated network component on slice L, and can be expressed as

$$\begin{aligned} d_{k,L}\left(t\right) = &\sum_{n\in\mathbf{N}}\sum_{m\in\mathbf{M}} \varphi^{BS}_{k,L,m}(t)\xi_{k,L,n}(t)d^{BS}_{k,L,m}(t) \\ &+ \sum_{n\in\mathbf{N}}\sum_{v\in\mathbf{V}} \varphi^{UAV}_{k,L,v}(t)\xi_{k,L,n}(t)d^{UAV}_{k,L,v}(t) \\ &+ \sum_{n\in\mathbf{N}}\sum_{o\in\mathbf{O}} \varphi^{LEO}_{k,L,o}(t)\xi_{k,L,n}(t)d^{LEO}_{k,L,o}(t). \end{aligned} \tag{22}$$

Finally, the average delay for slice L can be calculated as

$$D^{ave}_{L}\left(t\right) = \frac{1}{K_L}\sum_{k\in\mathbf{K_L}} D_{k,L}\left(t\right). \tag{23}$$

C. Wide-coverage slice

For the wide-coverage slice C, the MNOs aim to provide basic access services for as many users as possible. Consequently, the main objective of slice C is to maximize the number of users covered by the system, i.e., the coverage percentage. Here, we adopt the SINR-based coverage, which means if the SINR of user k exceeds the threshold γ_{th}, we say the user is in the coverage of the system. Then, the SINR of user k on slice C can be written as

$$\gamma_{k,C}\left(t\right) = \gamma^{BS}_{k,C}\left(t\right) + \gamma^{UAV}_{k,C}\left(t\right) + \gamma^{LEO}_{k,C}\left(t\right). \tag{24}$$

Accordingly, we can get the number of users covered by the system as

$$N^{cov}_{C}(t) = \sum_{k}^{K_C} \mathbb{1}\{\gamma_{k,C}(t) > \gamma_{th}\}. \tag{25}$$

Hereafter, the coverage percentage of slice C can be expressed as:

$$P^{cov}_{C}(t) = N^{cov}_{C}(t)/K_C \tag{26}$$

D. Multi-objective problem formulation

According to the above discussion, we can get the the MOOP for the three RAN slices to jointly optimize three different objectives of the throughput, the average delay and the coverage percentage. Then, by (20), (23) and (26), the MOOP is presented in (27)

$$\begin{aligned}
\mathbf{P}^*: & \max_{\{\boldsymbol{\Xi}_{\mathbf{s}},\boldsymbol{\Psi}_{\mathbf{s}},\mathbf{P}_{\mathbf{s}},\boldsymbol{\Lambda}_{\mathbf{UAV}}\}} \left\{ \begin{array}{l} R_H^{sum}(t),\ s=H, \\ -D_L^{ave}(t),\ s=L, \\ P_C^{cov}(t),\ s=C, \end{array} \right. \\
s.t.\ & C1: \xi_{k,s,n}(t), \varphi_{k,s,m(v,o)}(t) \in \{0,1\}, \forall k\in\mathbf{K}, n\in\mathbf{N}, \\
& \qquad \forall m\in\mathbf{M}, v\in\mathbf{V}, o\in\mathbf{O}, \\
& C2: \sum_{n\in\mathbf{N}} \sum_{\substack{m\in\mathbf{M} \\ (v\in\mathbf{V}, o\in\mathbf{O})}} \xi_{k,s,n}(t)\varphi_{k,s,m(v,o)}(t) \le 1, \forall k\in\mathbf{K}, \\
& C3: \sum_{m\in\mathbf{M}} \varphi_{k,s,m}^{BS}(t) + \sum_{v\in\mathbf{V}} \varphi_{k,s,v}^{UAV}(t) + \sum_{o\in\mathbf{O}} \varphi_{k,s,o}^{LEO}(t) \le 1, \forall k\in\mathbf{K}, \\
& C4: p_{k,s,m,n}^{BS}(t), p_{k,s,v,n}^{UAV}(t), p_{k,s,o,n}^{LEO}(t) \ge 0, \forall k\in\mathbf{K}, \\
& \qquad \forall n\in\mathbf{N}, m\in\mathbf{M}, v\in\mathbf{V}, o\in\mathbf{O}, \\
& C5: \sum_{s\in\{H,L,C\}} \sum_{k\in\mathbf{K}} \varphi_{k,s,m}^{BS}(t) \cdot \mathbb{1}\left\{\sum_{n\in\mathbf{N}} \xi_{k,s,n}(t)\right\} \le C_{th}, \\
& C6: d_{i,j}^{UAV}(t) \ge d_{\min}^{UAV}, \forall i,j\in\mathbf{V}, i\ne j,
\end{aligned} \tag{27}$$

where C1 shows the value range of binary variables $\xi_{k,s,n}(t)$ and $\varphi_{k,s,m(v,o)}(t)$, C2 exhibits that we can only assign at most one subcarrier to each user at time slot t, and C3 represents that a user is only associated with at most one network component at time slot t. Furthermore, C4 limits the non-negativity of the transmit power, while C5 shows the number of users associated with the same BS can not exceed the maximum capacity C_{th}. C6 illustrates that the distance between two UAVs is no less than d_{min}^{UAV}, where all UAVs are at the same altitude. In this model, we denote $\boldsymbol{\Psi}_{\mathbf{s}} = \left\{\boldsymbol{\Psi}_{\mathbf{s}}^{\mathbf{BS}}(t), \boldsymbol{\Psi}_{\mathbf{s}}^{\mathbf{UAV}}(t), \boldsymbol{\Psi}_{\mathbf{s}}^{\mathbf{LEO}}(t)\right\}$ and $\mathbf{P}_{\mathbf{s}} = \left\{\mathbf{P}_{\mathbf{s}}^{\mathbf{BS}}(t), \mathbf{P}_{\mathbf{s}}^{\mathbf{UAV}}(t), \mathbf{P}_{\mathbf{s}}^{\mathbf{LEO}}(t)\right\}$

V. Problem Transformation and Solution

In this section, we decompose the problem $\mathbf{P}^*$ into two sub-problems to reduce the complexity. Then, a two-layer MADDPG algorithm is proposed to solve the above problems.

A. Problem decomposition

We consider that users are preferentially associated to BSs, and the resources (bandwidth and power) of the TL and NTL do not overlapped, thus the problem $\mathbf{P}^*$ can be decoupled into the TL optimization sub-problem and the NTL sub-problem. Firstly, we will define the optimization objectives for the TL and NTL, respectively. The set of users on slice s associated with TL is denoted by $\mathbf{K}_{\mathbf{s}}^{\mathbf{T}}(t) = \{1, 2, \cdots, K_s^T(t), s \in \{H, L, C\}\}$. Consequently, the total number of users for slice s on NTL are defined as $K_s^{NT}(t) \in \mathbf{K}_{\mathbf{s}}^{\mathbf{NT}}(t)$. Similarly, each layer optimizes the three metrics: throughput, average delay, and coverage percentage. Then, the throughput of the TL and NTL are defined as following:

$$R_H^T(t) = \sum_{k\in\mathbf{K}_{\mathbf{H}}^{\mathbf{T}}(t)} R_{k,H}^{BS}(t), \tag{28}$$

and

$$R_H^{NT}(t) = \sum_{k\in\mathbf{K}_{\mathbf{H}}^{\mathbf{NT}}(t)} \left[R_{k,H}^{UAV}(t) + R_{k,H}^{LEO}(t)\right]. \tag{29}$$

Next, the average delay of the two layers are formulated as

$$\begin{aligned} D_L^T(t) = \frac{1}{K_L^T(t)} \sum_{k\in\mathbf{K}_{\mathbf{L}}^{\mathbf{T}}(t)} & \left\{ \frac{d_{k,L}^T(t)}{c} + \frac{A_{k,L}(t)}{R_{k,L}(t)} \right. \\ & \left. + \frac{\lambda_{k,L}(t)}{2\left[\left(R_{k,L}(t)\right)^2 - \lambda_2 R_{k,L}(t)\right]} \right\}, \end{aligned} \tag{30}$$

and

$$\begin{aligned} D_L^{NT}(t) = \frac{1}{K_L^{NT}(t)} \sum_{k\in\mathbf{K}_{\mathbf{L}}^{\mathbf{NT}}(\mathbf{t})} & \left\{ \frac{d_{k,L}^{NT}(t)}{c} + \frac{A_{k,L}(t)}{R_{k,L}(t)} \right. \\ & \left. + \frac{\lambda_{k,L}(t)}{2\left[\left(R_{k,L}(t)\right)^2 - \lambda_2 R_{k,L}(t)\right]} \right\}, \end{aligned} \tag{31}$$

where

$$d_{k,L}^T(t) = \sum_{n\in\mathbf{N}} \sum_{m\in\mathbf{M}} \varphi_{k,L,m}^{BS}(t)\xi_{k,L,n}(t)d_{k,L,m}^{BS}(t). \tag{32}$$

and

$$\begin{aligned} d_{k,L}^{NT}(t) = & \sum_{n\in\mathbf{N}} \sum_{v\in\mathbf{V}} \varphi_{k,L,v}^{UAV}(t)\xi_{k,L,n}(t)d_{k,L,v}^{UAV}(t) \\ & + \sum_{n\in\mathbf{N}} \sum_{o\in\mathbf{O}} \varphi_{k,L,o}^{LEO}(t)\xi_{k,L,n}(t)d_{k,L,o}^{LEO}(t). \end{aligned} \tag{33}$$

Finally, the coverage percentage of the two layers are respectively written as

$$P_C^T(t) = \frac{\sum_{k\in\mathbf{K}_{\mathbf{C}}^{\mathbf{T}}(t)} \mathbb{1}\{\gamma_{k,C}^{BS}(t) > \gamma_{th}\}}{K_C^T(t)} \tag{34}$$

and

$$P_C^{NT}(t) = \frac{\sum_{k\in\mathbf{K}_{\mathbf{C}}^{\mathbf{NT}}(t)} \mathbb{1}\{\gamma_{k,C}^{UAV}(t) + \gamma_{k,C}^{LEO}(t) > \gamma_{th}\}}{K_C^{NT}(t)} \tag{35}$$

Again, users are only associated with one network component and resources of the two layers are isolated, then, we have

$$\left\{ \begin{array}{l} R_H^{sum}(t) = R_H^T(t) + R_H^{NT}(t), \\ -D_L^{ave}(t) = -\left[D_L^T(t) + D_L^{NT}(t)\right], \\ P_C^{cov}(t) = P_C^T(t) + P_C^{NT}(t). \end{array} \right. \tag{36}$$

Thus, the problem $\mathbf{P}^*$ can be decoupled into the sub-problem $\mathbf{P_1}$ in the TL and $\mathbf{P_2}$ in the NTL, which can be constructed as

$$\begin{aligned} \mathbf{P_1}: & \max_{\{\mathbf{\Psi_s^{BS}},\mathbf{\Xi_s},\mathbf{P_s^{BS}}\}} \begin{cases} R_H^T(t), s=H \\ -D_L^T(t), s=L \\ P_C^T(t), s=C \end{cases} \\ s.t.\ & C1': \xi_{k,s,n}(t), \varphi_{k,s,m}^{BS}(t) \in \{0,1\}, \forall k \in \mathbf{K}, n \in \mathbf{N}, m \in \mathbf{M}, \\ & C2': \sum_{n\in\mathbf{N}} \sum_{m\in\mathbf{M}} \xi_{k,s,n}(t)\varphi_{k,s,m}(t) \le 1, \forall k \in \mathbf{K}, \\ & C4': p_{k,s,m,n}^{BS}(t) \ge 0, \forall k \in \mathbf{K}, n \in \mathbf{N}, m \in \mathbf{M}. \\ & C5': \sum_{s\in\{H,L,C\}} \sum_{k\in\mathbf{K}} \varphi_{k,s,m}^{BS}(t) \cdot \mathbb{1}\left\{\sum_{n\in\mathbf{N}} \xi_{k,s,n}(t)\right\} \le C_{th}, \end{aligned} \tag{37}$$

and

$$\begin{aligned} \mathbf{P_2}: & \max_{\left\{\begin{smallmatrix}\mathbf{\Lambda_{UAV}},\mathbf{\Psi_s^{UAV}},\mathbf{P_s^{UAV}} \\ \mathbf{\Psi_s^{LEO}},\mathbf{P_s^{LEO}},\mathbf{\Xi_s}\end{smallmatrix}\right\}} \begin{cases} R_H^{NT}(t), s=H \\ -D_L^{NT}(t), s=L \\ P_C^{NT}(t), s=C \end{cases} \\ s.t.\ & C1'': \xi_{k,s,n}(t), \varphi_{k,s,v}^{UAV}(t), \varphi_{k,s,o}^{LEO}(t) \in \{0,1\}, \\ & \qquad \forall k \in \mathbf{K_{NT}}, n \in \mathbf{N}, v \in \mathbf{V}, o \in \mathbf{O}, \\ & C2'': \sum_{n\in\mathbf{N}} \sum_{\substack{v\in\mathbf{V} \\ (o\in\mathbf{O})}} \xi_{k,s,n}(t)\varphi_{k,s,v(o)}(t) \le 1, \forall k \in \mathbf{K_{NT}}, \\ & C3'': \sum_{v\in\mathbf{V}} \varphi_{k,s,v}^{UAV}(t) + \sum_{o\in\mathbf{O}} \varphi_{k,s,o}^{LEO}(t) \le 1, \forall k \in \mathbf{K_{NT}}, \\ & C4'': p_{k,s,v,n}^{UAV}(t), p_{k,s,o,n}^{LEO}(t) \ge 0, \forall k \in \mathbf{K_{NT}}, n \in \mathbf{N}, \\ & \qquad \forall v \in \mathbf{V}, o \in \mathbf{O}, \\ & C5: d_{i,j}^{UAV}(t) \ge d_{\min}^{UAV}, \forall i,j \in \mathbf{V}, i \ne j, \end{aligned} \tag{38}$$

where $\mathbf{K_{NT}} = \left\{\mathbf{K_s^{NT}}(t)\right\}$ is the set of users unserved by TL, including the offloaded and unconnected users. To be specific, the sub-problem $\mathbf{P_1}$ firstly optimizes the user-BS association and resource allocation for all users at TL. However, due to the limited capacity and resources of TL, the surpassing users will be offloaded to the NTL, and there are also users who cannot access the TL[1]. Therefore $\mathbf{P_2}$ is formulated to serve these users.

B. The two-layer MADDPG Model Structure

Considering the relationship among the above two decoupled sub-problems, a two-layer MADDPG model is proposed as shown in Fig. 2. Specifically, the fist layer MADDPG consisting of three agents is used to optimize the user-BS association and resource allocation for three slices in the TL. Then, the second layer MADDPG with three agents optimizes the user-UAV and user-LEO satellite association and resource allocation for the unserved users on each slice in the NTL. In this way, the two sub-problems are optimized iteratively by the two-layer MADDPG. The three agents in each layer corresponding to three RAN slice s, termed as Agent $l_{i,s}$ for layer i, $\forall i \in \{1,2\}, s \in \{H,L,C\}$.

Firstly, we define the four-tuples of Markov decision process $\left\langle \mathbf{O}, \mathbf{A}, \mathbf{R}, \mathbf{O}' \right\rangle$ for the system, which represent the set of state observation, the action space, reward functions and the next state observation, respectively. Then, we can get the four-tuples for the first layer MADDPG as below.

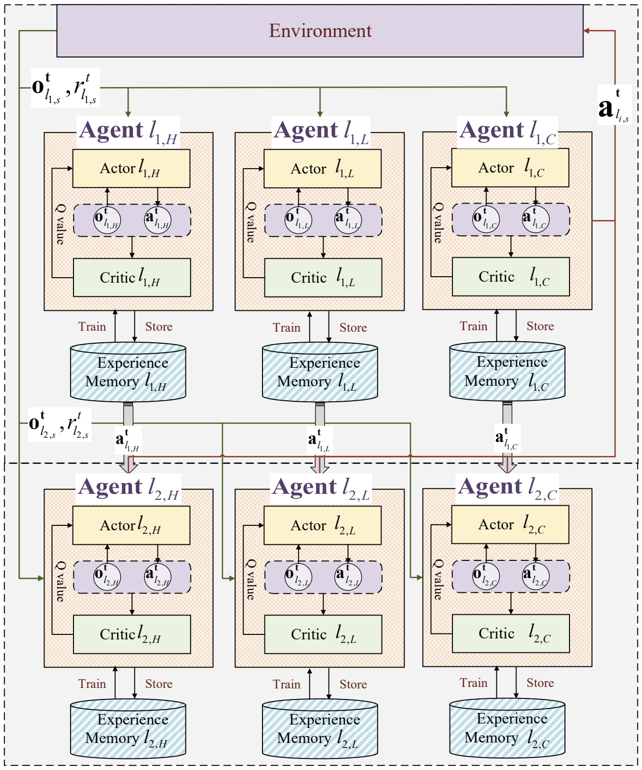

Fig. 2. The structure of the proposed two-layer MADDPG algorithm.

State $\mathbf{o_{1,s}^t}$ and the next state $\mathbf{o_{1,s}^{t+1}}$ of Agent $l_{1,s}$: The state observation is mapped to an environment feature vector of channel states, users' requests and users' positions. Therefore, the state observation of Agent $l_{1,s}$ at the time slot t is defined:

$$\mathbf{o_{1,s}^t} = \left\{\mathbf{h_s^{BS}(t)}, \mathbf{A_L}(\mathbf{t}), \mathbf{\Lambda_{UE}(t)}\right\}, \tag{39}$$

where $\mathbf{h_s^{BS}(t)}$ is the set of the channel coefficient $h_{k,s,m,n}^{BS}(t)$.

Obviously, the next state observation of Agent $l_{1,s}$ at the time slot $t+1$ can be formulated as:

$$\mathbf{o_{1,s}^{t+1}} = \left\{\mathbf{h_s^{BS}(t+1)}, \mathbf{A_L}(\mathbf{t+1}), \mathbf{\Lambda_{UE}(t+1)}\right\}. \tag{40}$$

Action $\mathbf{a_{1,s}^t}$ of Agent $l_{1,s}$: At the first layer, each slice needs to optimize the user-BS association, subcarrier allocation and power control for all users according to the $\mathbf{o_{1,s}^t}$ at the time slot t. Hence, the set of the actions can be denoted as:

$$\mathbf{a_{1,s}^t} = \left\{\mathbf{\Psi_s^{BS}}, \mathbf{\Xi_s}, \mathbf{P_s^{BS}}\right\}. \tag{41}$$

Here, we relax the binary variables in $\mathbf{\Psi_s^{BS}}$ and $\mathbf{\Xi_s}$ to continuous variables with range of [0,1], which matches the continuous action space in MADDPG algorithm. The variables finally are converted back to binary variables by a rounding function.

[1] They are called offloaded users and unconnected users, respectively, or collectively referred to as unserved users

Reward $r_{1,s}^{t}$ of agent $l_{1,s}$: According to (37), the reward of Agent $l_{1,s}$ is expressed as follows:

$$r_{1,s}^{t}=\begin{cases}\overline{R_{H}^{T}}\left(t\right),s=H,\\ \overline{\beta_{1}-D_{L}^{T}\left(t\right)},s=L,\\ \overline{P_{C}^{T}}\left(t\right),s=C,\end{cases}\tag{42}$$

where β_{1} is to guarantee the non-negativity of the delay objective for the first layer. Furthermore, we normalize the reward for each Agent $l_{1,s}$ by 0-1 normalization to improve the convergence rate of the model.

Similarly, we define the four-tuples for the second layer MADDPG.

State $\mathbf{o_{2,s}^{t}}$ and the next state $\mathbf{o_{2,s}^{t+1}}$ of Agent $l_{2,s}$: The state space of Agent $l_{2,s}$ includes the environment feature vector of channel states, data arrivals and users' position, and the unserved users. Therefore, the state observation of Agent $l_{2,s}$ at the time slot t is defined as:

$$\begin{aligned}\mathbf{o_{2,s}^{t}}=&\Big\{\mathbf{\widehat{h}_{s}^{UAV}(t)},\mathbf{\tilde{h}_{s}^{UAV}(t)},\mathbf{h_{s}^{LEO}(t)},\\&\mathbf{A_{L}}\left(\mathbf{t}\right),\mathbf{\Lambda_{UE}(t)},\mathbf{K_{NT}(t)}\Big\},\end{aligned}\tag{43}$$

where $\mathbf{\widehat{h}_{s}^{UAV}(t)}=\left\{\widehat{h}_{k,s,v,n}^{UAV}(t)\right\}$, $\mathbf{\tilde{h}_{s}^{UAV}(t)}=\left\{\tilde{h}_{k,s,v,n}^{UAV}(t)\right\}$ and $\mathbf{h_{s,i}^{UAV}(t)}=\left\{h_{k,s,o,n}^{LEO}(t)\right\}$ are the set of the channel coefficient for the users on slice s.

Obviously, the next state observation of Agent $l_{2,s}$ at the time slot $t+1$ can be formulated as:

$$\begin{aligned}\mathbf{o_{2,s}^{t+1}}=&\Big\{\mathbf{\widehat{h}_{s}^{UAV}(t+1)},\mathbf{\tilde{h}_{s}^{UAV}(t+1)},\mathbf{h_{s}^{LEO}(t+1)},\\&\mathbf{A_{L}}\left(\mathbf{t+1}\right),\mathbf{\Lambda_{UE}(t+1)},\mathbf{K_{NT}(t+1)}\Big\}.\end{aligned}\tag{44}$$

Action $\mathbf{a_{2,s}^{t}}$ of Agent $l_{2,s}$: the second MADDPG needs to optimize the positions of the UAVs, the user-UAV or user-LEO satellite association, subcarrier allocation and power control for the offloaded and unconnected users at the time slot t. Hence, the set of the actions can be denoted as:

$$\mathbf{a_{2,s}^{t}}=\left\{\mathbf{\Lambda_{UAV}},\mathbf{\Psi_{s}^{UAV}},\mathbf{P_{s}^{UAV}},\mathbf{\Psi_{s}^{LEO}},\mathbf{P_{s}^{LEO}},\mathbf{\Xi_{s}}\right\}.\tag{45}$$

Also, the binary variables in $\mathbf{a_{2,s}^{t}}$ are relaxed to continuous variables with range of [0,1].

Reward $r_{2,s}^{t}$ of agent $l_{2,s}$: According to (38), the reward of Agent $l_{2,s}$ is as follows:

$$r_{2,s}^{t}=\begin{cases}\overline{R_{H}^{NT}}\left(t\right),s=H,\\ \overline{\beta_{2}-D_{L}^{NT}\left(t\right)},s=L,\\ \overline{P_{C}^{NT}}\left(t\right),s=C,\end{cases}\tag{46}$$

where β_{2} is to guarantee the non-negativity of the delay objective for the second layer.

C. The two-layer MADDPG algorithm

In the proposed two-layer MADDPG model, all the agents rely on the actor-critic(AC) structure, which contains an actor and a critic, as seen in Fig. 2. Concretely, the actor selects an action according to the current state; then the critic evaluates the choose action and returns a Q-value; finally the actor modifies following action selection policies with the returned Q-value. The policy network $\mu_{\theta_s}\left(\mathbf{o_{s}^{t}};\theta^{\mu_s}\right)$ is referred to an actor network named as actor s, to maximize the long-term cumulative discounted reward:

$$J_{s}\left(\mu\right)=E_{\mu}\left[\sum_{i=0}^{T-1}\gamma^{i}r_{s}^{i}\right],\tag{47}$$

where γ is the discount factor.

While the action-value function refers to an critic network (named critic s), is defined by Bellman's equation as:

$$\begin{aligned}&Q_{\theta_s}\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s}\right)\\&=r_{s}^{t}+\gamma\max_{\mathbf{a}}Q_{\theta_s}\left(\mathbf{o_{s}^{t+1}},\mathbf{a_{s}^{t+1}},\mathbf{a_{S\backslash s}^{t+1}};\theta^{Q_s}\right),\end{aligned}\tag{48}$$

Further, the policy network is updated by the deterministic policy gradient (DPG) algorithm, as follows:

$$\begin{aligned}\nabla_{\theta^{\mu_s}}J_{s}=&E_{\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}}\sim\mathbf{D_{s}}}\left[\nabla_{\theta^{\mu_s}}\mu_{\theta_s}\left(\mathbf{o_{s}^{t}};\theta^{\mu_s}\right)\right.\\&\left.\cdot\nabla_{\mathbf{a_{s}^{t}}}Q_{\theta_s}\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s}\right)|_{\mathbf{a_{s}^{t}}=\mu_{\theta_s}\left(\mathbf{o_{s}^{t}};\theta^{\mu_s}\right)}\right],\end{aligned}\tag{49}$$

where $\mathbf{D_{s}}$ is the experience memory of the agent s.

Then, we update the critic network by minimizing the loss function

$$\begin{aligned}L_{s}=&E_{\mathbf{D_{s}}}\left[\left(r_{s}^{t}+\gamma Q_{\theta_s}'\left(\mathbf{o_{s}^{t+1}},\mu_{\theta_s}'\left(\mathbf{o_{s}^{t+1}};\theta^{\mu_s'}\right);\theta^{Q_s'}\right)\right.\right.\\&\left.\left.-Q_{\theta_s}\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s}\right)\right)^{2}\right],\end{aligned}\tag{50}$$

where $\theta^{\mu_s'}$ and $\theta^{Q_s'}$ represent the parameters of the target actor and the target critic, respectively.

Then, the gradient of the critic network is shown below:

$$\begin{aligned}\nabla_{\theta^{Q_s}}L_{s}=&E_{\mathbf{D_{s}}}\left[\left(r_{s}^{t}+\gamma Q_{\theta_s}'\left(\mathbf{o_{s}^{t+1}},\mu_{\theta_s}'\left(\mathbf{o_{s}^{t+1}};\theta^{\mu_s'}\right);\theta^{Q_s'}\right)\right.\right.\\&\left.\left.-Q_{\theta_s}\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s}\right)\right)\nabla_{\theta^{Q_s}}Q_{\theta_s}\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s}\right)\right].\end{aligned}\tag{51}$$

Furthermore, the MADDPG algorithm introduces two deep neural networks (DNNs) for each actor and critic: online network and target network, to promise the stability of the training process, i.e.,

$$\text{actor}\begin{cases}\text{online}:\mu_{\theta_s}\left(\mathbf{o_{s}^{t}};\theta^{\mu_s}\right),\text{update }\theta^{\mu_s},\\ \text{target}:\mu_{\theta_s}'\left(\mathbf{o_{s}^{t}};\theta^{\mu_s'}\right),\text{update }\theta^{\mu_s'},\end{cases}\tag{52}$$

and

$$\text{critic}\begin{cases}\text{online}:Q_{\theta_s}\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s}\right),\text{update }\theta^{Q_s},\\ \text{target}:Q_{\theta_s}'\left(\mathbf{o_{s}^{t}},\mathbf{a_{s}^{t}},\mathbf{a_{S\backslash s}^{t}};\theta^{Q_s'}\right),\text{update }\theta^{Q_s'}.\end{cases}\tag{53}$$

Based on the above discussion, the two sub-problems can be solved by the proposed two-layer MADDPG algorithm shown in Algorithm 1. To elaborate, the first layer MADDPG algorithm is used to optimize the user-BS association and resource allocation for each associated users at each time slot. Then, if there are BSs overloaded, the LB scheme will be triggered disconnect the user from the BS according to the offload priority of the slice. Next, the second layer MADDPG algorithm is responsible for the user-UAV or -LEO satellite association and resource allocation for the unserved users.

Algorithm 1 The proposed two layer MADDPG Algorithm.

```
Initialization:
    Initialize critic networks and actor networks of the two-layer algorithms, respectively.
    Initialize the experience memory $\mathbf{D}_{l_i}$.
    Obtain the initial set of states $\mathbf{O}_{l_i}$.
For Episode $t$ = 1, ..., $E$ do:
    Obtain the initial state space for the first layer.
    For Step $t$ = 1, ..., $T$ do:
        For the layer $l_1$, select action $\mathbf{a^t_{1,s}} = \mu_\theta\left(\mathbf{o^t_{1,s}}; \theta^\mu\right) + \mathbf{N_t}$ based on current policy and exploration noise.
            Count the load $LO_n$ for each BS, if $LO_n > C_{th}$, $LO_n - C_{th}$ users will be randomly selected to disconnected with BS $n$ from the connected users on lower-priority slices, i.e., the corresponding association indicators in $\mathbf{a^t_{1,s}}$ are set to 0.
        Perform actions $\mathbf{a^t_{1,s}}$ and get the rewards of $r^t_{1,s}$ as well as the new states $\mathbf{o^{t+1}_{1,s}}$.
        Transfer $\mathbf{a^t_{1,s}}$ and $r^t_{1,s}$ to Agent $l_{2,s}$.
        Calculate $\mathbf{K_{NT}}(t)$ according to $\mathbf{a^t_{1,s}}$.
        Get the state space $\mathbf{o^t_{2,s}}$ for the second layer.
        Agent $l_{2,s}$ selects action $\mathbf{a^t_{2,s}}$ based on $\mathbf{o^t_{2,s}}$.
        Perform actions $\mathbf{a^t_{2,s}}$ and obtain the rewards $r^t_{2,s}$ as well as the new states $\mathbf{o^{t+1}_{2,s}}$.
        Store the four-tuples of $\left\langle \mathbf{o^t_{1,s}}, \mathbf{a^t_{1,s}}, r^t_{1,s}, \mathbf{o^{t+1}_{1,s}} \right\rangle$ in the experience memory $\mathbf{D_{1,s}}$.
        Store the four-tuples of $\left\langle \mathbf{o^t_{2,s}}, \mathbf{a^t_{2,s}}, r^t_{2,s}, \mathbf{o^{t+1}_{2,s}} \right\rangle$ in the experience memory $\mathbf{D_{2,s}}$.
        Sample random mini-batches from $\mathbf{D_{1,s}}$ and $\mathbf{D_{2,s}}$, respectively.
        Update the actors by using the gradient policy algorithm of (49).
        Update the critics by minimizing loss function of (50).
        Update the parameters of the target networks
            $\begin{cases} \theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\,\theta^{Q'}, \\ \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\,\theta^{\mu'}. \end{cases}$
        $\mathbf{o^t_{1,s}} \leftarrow \mathbf{o^{t+1}_{1,s}}$, $\mathbf{o^t_{2,s}} \leftarrow \mathbf{o^{t+1}_{2,s}}$.
    End for
End for
```

VI. Performance Evaluation

Here we report extensive simulation results for validating the theoretical analysis and for comparing the performance of the proposed solution with the benchmark algorithms. The experience memory of each agent is set with 2000 four-tuples, and D_{mini}=32 mini-batches are sampled for training. Referring to [5], [35], the users are randomly distributed with the terrestrial area 3000×3000 (m^2), and a 3D Euclidean coordinate model is adopted. The number of BSs and UAVs are set as $M = 2$, $V = 2$ and $O = 1$, where the coordinates of the terrestrial BSs are fixed as (1000,1000,0) (m) and (2000,2000,0) (m), respectively. Besides, the altitudes of the UAVs and of the LEO satellites are fixed as $z_v^{UAV}(t) = 100m$ and $z_0^{LEO}(t) = 200,000m$, respectively. The available bandwidth for each type of network components is the same as B=30MHz, which is divided into $N = 10$ subcarriers. Unless otherwise stated, the other parameters are given as follows: P_B = 10dBW, P_V = 20dBW, P_O = 30dBW, $\delta = 0.1$s/slot, $N_0 = -130$dBm/Hz, $\beta_1 = \beta_2 = 0.1$s, $\alpha = 1.5$, $R = 6$, $h_0 = -30$dB, $d_{min}^{UAV} = 100$m, $\lambda_{k,L} = 50$kbits/slot. In order to reflect the advantages of our proposed schemes, we give the following benchmark schemes for comparison:

- Benchmark 1, "Load Balancing" method: all the users have the same priority independently on their employed slice, which is, when the BS is overload, all connecting users are randomly disconnected with the BS.
- Obviously, Benchmark 2 of the scheme "Without Load Balancing": we have no further operation for the offloaded BSs.
- Then, in Benchmark 3 and 4, we consider the conventional single-layer "MADDPG" and "MAPPO" algorithm relying on six parallel distributed Agents.
- Finally, we compare with Benchmark 5: "fixed UAVs" scheme, where the positions of the UAVs are fixed at (1000, 2000, 100) (m), and (2000, 1000, 100) (m), respectively; while in our scheme, the positions are optimized in $\mathbf{P_2}$.

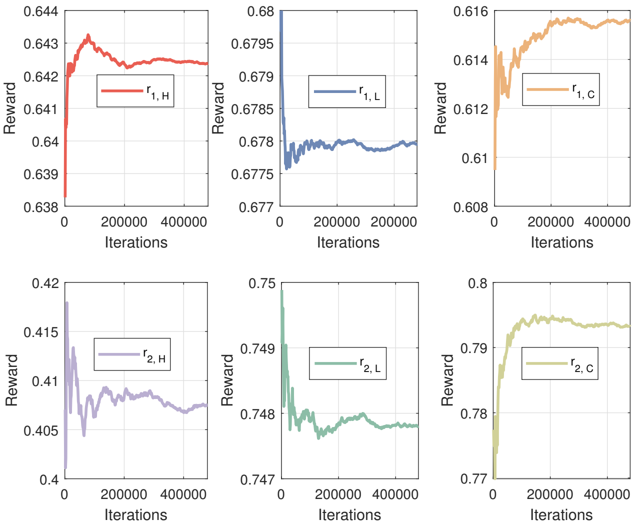

Fig. 3. Convergence of the two-layer MADDPG algorithm.

Fig. 3 shows the convergence performance of our two-layer MADDPG algorithm. We can see that all rewards converge to a relatively stable value after about 30000 training iterations. Though, there are still some small fluctuations, because the exploration probability $\mathbf{N_t}$ is added when choosing actions and the three slices are competing for the resources. In addition, the reward $r_{1,H}$ of Layer 1 is larger than $r_{2,H}$ of Layer 2, which corresponds to the fact that the high-throughput slice, which has the highest priority to access the TL. Similarly, the reward $r_{1,C}$ of Layer 1 is smaller than $r_{2,C}$ of Layer 2, as the mMTC slice has the lowest priority to access TL. And, $r_{1,L}$ is the same as $r_{2,L}$, because the slice L is in the middle priority.

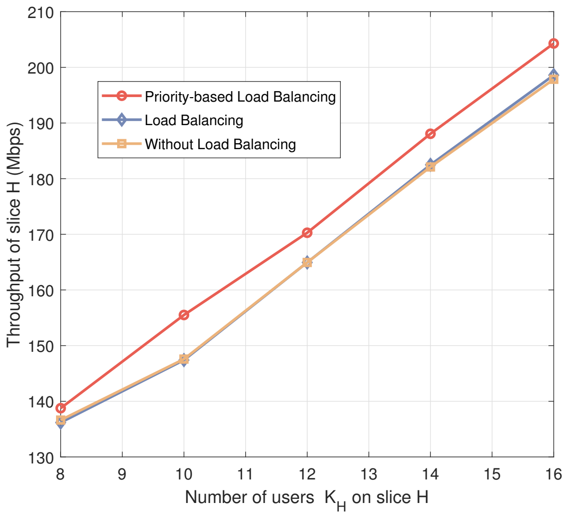

Fig. 4. Throughput versus the number of users K_U of slice U.

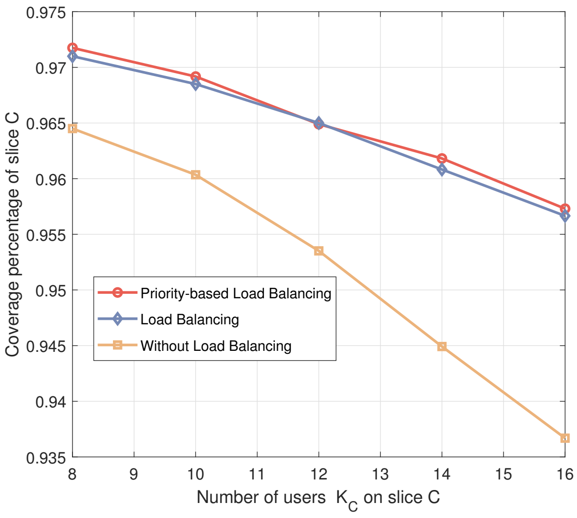

Fig. 6. Coverage percentage versus the number of users K_C of slice C.

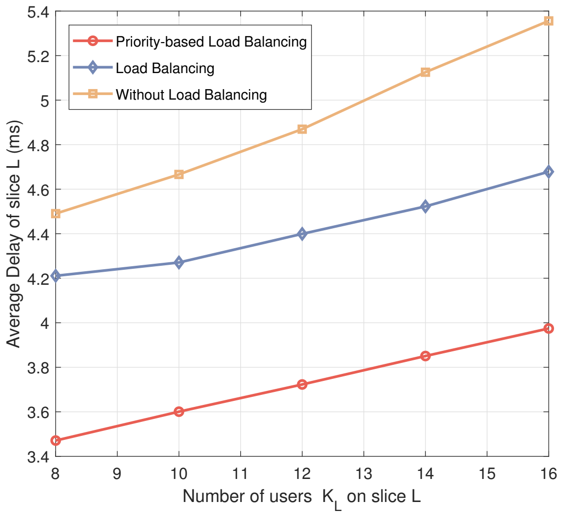

Fig. 5. Delay versus the number of users K_L of slice L.

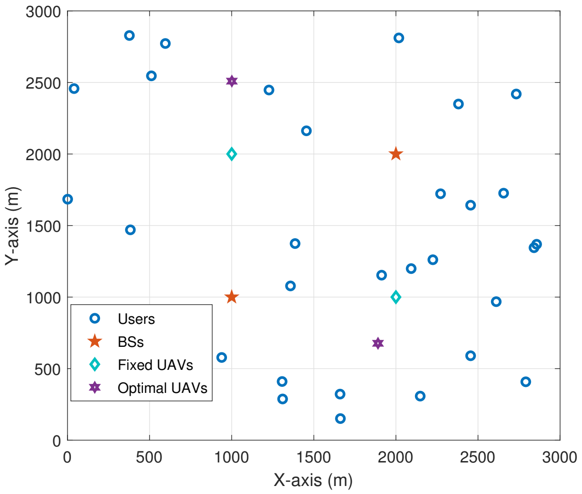

Fig. 7. Coordinate distribution of users, BSs, fixed UAVs and optimal UAVs ($K1 = K2 = K3 = 10$).

Fig. 4, Fig. 5, and Fig. 6 characterize the performance versus the number of users on each slice, while comparing our original proposal with the Benchmarks 1 and 2. It can be observed that all the performance metrics of the proposed scheme are better than that of the benchmark schemes. Among the two LB schemes, we experience a significant enhancement for average delay of slice L in our proposed scheme, though the throughput of slice H and the coverage percentage of slice C are almost the same for the two schemes. For the three schemes, the users on slice L have the lowest probability to access the NTL in our proposed scheme, the long distance transmission will lead to a higher delay. The scheme of “Without Load Balancing” has relatively low performance over the system. Compared with the load balancing schemes, “Without Load Balancing” has no limit on the number of users served by BSs, then the interference is larger, which leads to decline in performance. Considering the trends for the three schemes of Fig. 4, Fig. 5, and Fig. 6, we can see that both the throughput and the average delay of all schemes are increasing with the increasing of K_s, whilst the coverage percentage is decreasing.

Fig. 7 compares the two-dimensional coordinates of the UAVs in our proposed scheme to that of the fixed UAVs’ position scheme, where users are randomly distributed in the area. Naturally, the optimal UAVs will actively approach the area of high-density users to reduce the transmission distance, while far away from the BSs to serve the area that BSs cannot covered or served. Thereby, the performance of the system can be enhanced, and it can be concluded that our scheme can provide a reasonable network component deployment.

Fig. 8 and Fig. 9 illustrate the loads of each network component versus the number of users, in which the capacity of each BS is set to 10. In order to facilitate the analysis, we give the average of 5 experiment trials with the same parameter

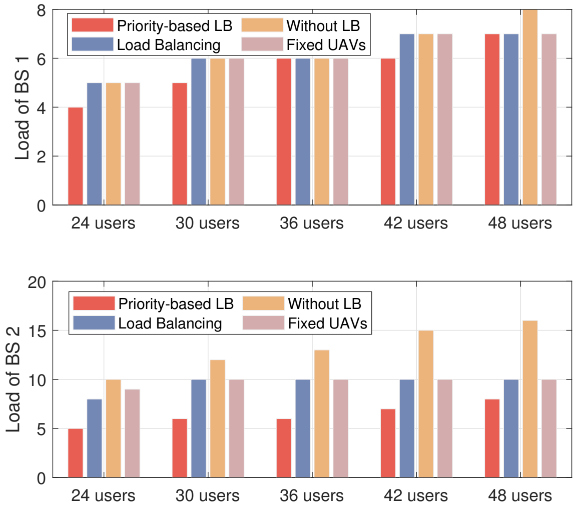

Fig. 8. Load of each BS with different number of users for different schemes).

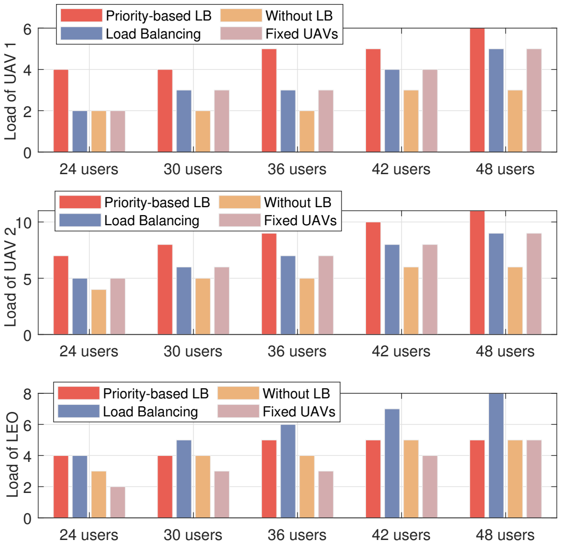

Fig. 9. Load of UAVs and LEO satellite with different number of users for different schemes).

configuration. As we consider LB of the BSs in the TL, the overloaded BSs will release part of their loads to the NTL by using the proposed LB algorithm. Observe from the Fig. 8; with the increase of the users, the load of BS2 is increasing, and overloads the capacity of BS 2 when the number of users exceeds 30. Then, the more users access to the same BS, the higher interference will be generated and less resource will be allocated, and the performance thus decrease (which can see in Fig. 4, Fig. 5, and Fig. 6). Thus, the load balancing is necessary and should be taken. Though we consider the BS load balancing to simplify the analysis, the UAVs and LEO satellites load balancing may be studied in the future.

Fig. 10, Fig. 11, and Fig. 12 display the variation of throughput, average delay, and coverage percentage versus the number

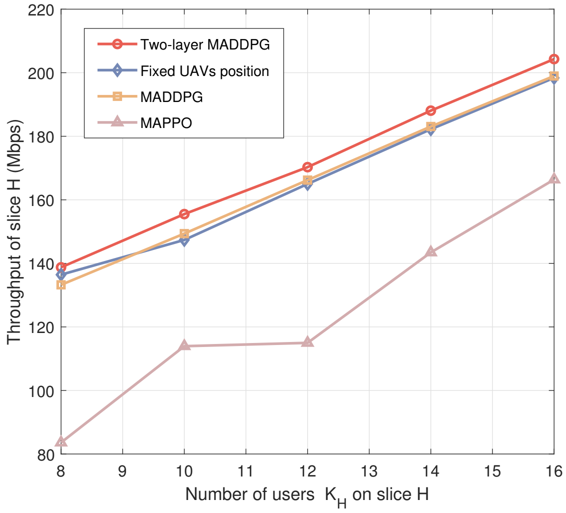

Fig. 10. Throughput comparison with the benchmarkers.

of users for corresponding slices with different algorithms. Firstly, it can be seen that all the performance metrics of the proposed algorithm are better than those of the fixed UAVs' position algorithm, MADDPG algorithm and MAPPO algorithm. Explicitly, the difference of throughput performance between the proposed algorithm and the fixed UAVs' position algorithm are relatively small, while their average delays are quite different. This reveals that the communication distance has an impact on the average delay. In addition, as our settings of the fixed UAVs' positions are not very far from the optimal positions, it has a small impact on the performance of throughput and coverage percentage. As expected, since we apply a two-layer resource allocation mechanism, the proposed algorithm exhibits higher performance advantage than the single-layer option. For the DRL, it will reach a higher collision probability due to a large action space, which will gravely impact both the performance of algorithm and the QoS of users.

VII. Conclusive Remarks

In this article, we propose an original priority-based cross-layer LB scheme for SAGIN slicing. Firstly, high-throughput, low-delay, and wide-coverage slices are considered on top of the same physical SAGIN. Then, the throughput, average delay, and coverage percentage of the three slices are jointly optimized with a priority-based LB manner. In this manner, two kinds of priorities are considered: association priority and offloading priority. Specifically, the former means that users has the priority access to the BSs in the TL, while the latter sets different offloading priorities for each slice (users on low-priority slices will be offloaded to the NTL first). Then, the optimization problem is formulated and decoupled into two sub-problems in the TL and NTL, respectively. Finally, we propose a novel two-layer MADDPG algorithm for solving the above sub-problems; the first layer MADDPG optimizes the user-BS association and resource allocation at the TL; the second-layer MADDPG determines the most suitable UAVs'

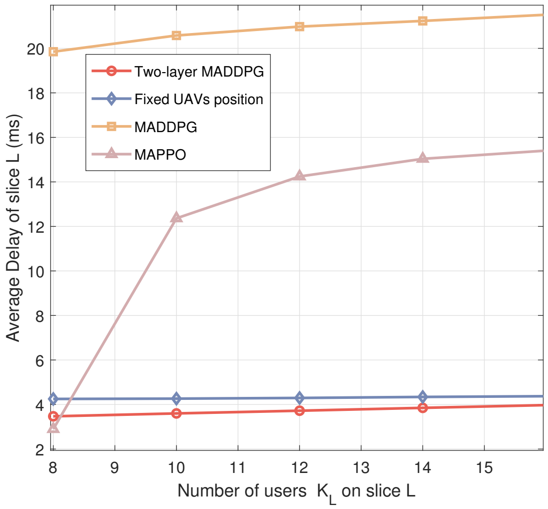

Fig. 11. Average delay comparison with the benchmarkers.

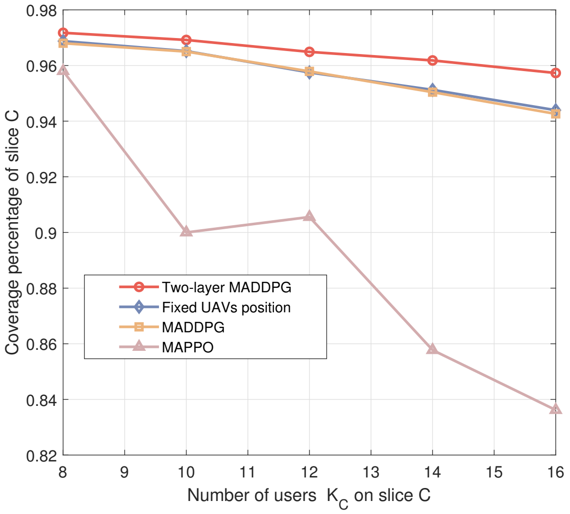

Fig. 12. Coverage percentage comparison with the benchmarkers.

position deployment, user-UAV/LEO satellite association, and resource allocation for the unserved users at the NTL.

The encouraging results already achieved and presented in this paper are stimulating some additional research work. As the current paper aims to the RAN domain, the end-to-end SAGIN slicing is the next step for our ongoing and future research work, where we have already preliminary results that a similar holistic approach can gain significant performance improvements as well.

References

[1] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, D. Niyato, O. Dobre, and H. V. Poor, "6G Internet of Things: A Comprehensive Survey," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 359–383, 2022.

[2] Z. Jia, M. Sheng, J. Li, and Z. Han, "Toward Data Collection and Transmission in 6G Space–Air–Ground Integrated Networks: Cooperative HAP and LEO Satellite Schemes," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 10 516–10 528, 2022.

[3] J. Liu, X. Du, J. Cui, M. Pan, and D. Wei, "Task-Oriented Intelligent Networking Architecture for the Space–Air–Ground–Aqua Integrated Network," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5345–5358, 2020.

[4] H. Cao, J. Du, H. Zhao, D. X. Luo, N. Kumar, L. Yang, and F. R. Yu, "Toward Tailored Resource Allocation of Slices in 6G Networks With Softwarization and Virtualization," *IEEE Internet of Things Journal*, vol. 9, no. 9, pp. 6623–6637, 2022.

[5] G. Zhou, L. Zhao, G. Zheng, S. Song, J. Zhang, and L. Hanzo, "Multiobjective Optimization of Space-Air-Ground Integrated Network Slicing Relying on a Pair of Central and Distributed Learning Algorithms," *IEEE Internet of Things Journal*, pp. 1–1, 2023.

[6] A. M. Seid, H. N. Abishu, A. Erbad, and C. F. Chiasserini, "Hierarchical DRL-empowered Network Slicing in Space-Air-Ground Networks," in *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*, 2023, pp. 4680–4685.

[7] H. Qu, Y. Luo, J. Zhao, and Z. Luan, "An lbmre-olsr routing algorithm under the emergency scenarios in the space-air-ground integrated networks," in *2020 Information Communication Technologies Conference (ICTC)*, 2020, pp. 103–107.

[8] T. Sun and L. Shi, "Load Balancing and Carrier-to-Noise Ratio based Handover Algorithm for LEO Satellite Network," in *2023 IEEE 11th International Conference on Information, Communication and Networks (ICICN)*, 2023, pp. 207–211.

[9] M. Tayyab, X. Gelabert, and R. Jäntti, "A Survey on Handover Management: From LTE to NR," *IEEE Access*, vol. 7, pp. 118 907–118 930, 2019.

[10] J. Liu, R. Luo, T. Huang, and C. Meng, "A Load Balancing Routing Strategy for LEO Satellite Network," *IEEE Access*, vol. 8, pp. 155 136–155 144, 2020.

[11] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-Air-Ground Integrated Network: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2714–2741, 2018.

[12] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo, "A Survey of Multi-Objective Optimization in Wireless Sensor Networks: Metrics, Algorithms, and Open Problems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 550–586, 2017.

[13] P. Zhang, Y. Zhang, N. Kumar, and C.-H. Hsu, "Deep Reinforcement Learning Algorithm for Latency-Oriented IIoT Resource Orchestration," *IEEE Internet of Things Journal*, vol. 10, no. 8, pp. 7153–7163, 2023.

[14] C. Zhou, W. Wu, H. He, P. Yang, F. Lyu, N. Cheng, and X. Shen, "Deep Reinforcement Learning for Delay-Oriented IoT Task Scheduling in SAGIN," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 911–925, 2021.

[15] J. Cui, S. X. Ng, D. Liu, J. Zhang, A. Nallanathan, and L. Hanzo, "Multiobjective Optimization for Integrated Ground-Air-Space Networks: Current Research and Future Challenges," *IEEE Vehicular Technology Magazine*, vol. 16, no. 3, pp. 88–98, 2021.

[16] R. Lowe, Y. WU, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[17] X. You, C.-X. Wang, J. Huang, X. Gao, Z. Zhang, M. Wang, Y. Huang, C. Zhang, Y. Jiang, J. Wang *et al.*, "Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts," *Science China Information Sciences*, vol. 64, pp. 1–74, 2021.

[18] S. Mahboob and L. Liu, "Revolutionizing Future Connectivity: A Contemporary Survey On AI-Empowered Satellite-Based Non-Terrestrial Networks in 6G," *IEEE Communications Surveys & Tutorials*, pp. 1–1, 2024.

[19] L. Zhang, W. Abderrahim, and B. Shihada, "Heterogeneous Traffic Offloading in Space-Air-Ground Integrated Networks," *IEEE Access*, vol. 9, pp. 165 462–165 475, 2021.

[20] H. Cao, S. Shen, Y. Guo, S. Wu, H. Zhang, and P. Zhang, "Resource Allocation and Orchestration of Slicing Services in Softwarized Space-Aerial-Ground Integrated Networks," in *2023 International Wireless Communications and Mobile Computing (IWCMC)*, 2023, pp. 769–774.

[21] H. Cao, S. Garg, G. Kaddoum, M. Alrashoud, and L. Yang, "Efficient Resource Allocation of Slicing Services in Softwarized Space-Aerial-Ground Integrated Networks for Seamless and Open Access Services," *IEEE Transactions on Vehicular Technology*, pp. 1–13, 2023.

[22] A. Asheralieva, D. Niyato, and X. Wei, "Ultrareliable Low-Latency Slicing in Space–Air–Ground Multiaccess Edge Computing Networks for Next-Generation Internet of Things and Mobile Applications," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 3956–3978, 2024.

[23] T. Sun and L. Shi, "Load Balancing and Carrier-to-Noise Ratio based Handover Algorithm for LEO Satellite Network," in *2023 IEEE 11th*

International Conference on Information, Communication and Networks (ICICN), 2023, pp. 207–211.

[24] Y. Wang, J. Che, N. Wang, L. Liu, N. Wu, X. Zhong, and X. Han, “Load-Balancing Method for LEO Satellite Edge-Computing Networks Based on the Maximum Flow of Virtual Links,” *IEEE Access*, vol. 10, pp. 100 584–100 593, 2022.

[25] P. Zuo, C. Wang, Z. Wei, Z. Li, H. Zhao, and H. Jiang, “Deep Reinforcement Learning Based Load Balancing Routing for LEO Satellite Network,” in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, 2022, pp. 1–6.

[26] J. Tao, S. Liu, and C. Liu, “A Traffic Scheduling Scheme for Load Balancing in SDN-Based Space-Air-Ground Integrated Networks,” in *2022 IEEE 23rd International Conference on High Performance Switching and Routing (HPSR)*, 2022, pp. 95–100.

[27] F. Tang, H. Hofner, N. Kato, K. Kaneko, Y. Yamashita, and M. Hangai, “A Deep Reinforcement Learning-Based Dynamic Traffic Offloading in Space-Air-Ground Integrated Networks (SAGIN),” *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 276–289, 2022.

[28] Z. Hu, F. Zeng, Z. Xiao, B. Fu, H. Jiang, H. Xiong, Y. Zhu, and M. Alazab, “Joint Resources Allocation and 3D Trajectory Optimization for UAV-Enabled Space-Air-Ground Integrated Networks,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 11, pp. 14 214–14 229, 2023.

[29] P. Zhang, Y. Su, J. Wang, C. Jiang, C.-H. Hsu, and S. Shen, “Reinforcement learning assisted bandwidth aware virtual network resource allocation,” *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4111–4123, 2022.

[30] S. Zhang, A. Liu, C. Han, X. Liang, X. Xu, and G. Wang, “Multiagent Reinforcement Learning-Based Orbital Edge Offloading in SAGIN Supporting Internet of Remote Things,” *IEEE Internet of Things Journal*, vol. 10, no. 23, pp. 20 472–20 483, 2023.

[31] A. Paul, K. Singh, M.-H. T. Nguyen, C. Pan, and C.-P. Li, “Digital Twin-assisted Space-Air-Ground Integrated Networks for Vehicular Edge Computing,” *IEEE Journal of Selected Topics in Signal Processing*, pp. 1–16, 2023.

[32] S. Mao, S. He, and J. Wu, “Joint UAV Position Optimization and Resource Scheduling in Space-Air-Ground Integrated Networks With Mixed Cloud-Edge Computing,” *IEEE Systems Journal*, vol. 15, no. 3, pp. 3992–4002, 2021.

[33] J. Ye, S. Dang, B. Shihada, and M.-S. Alouini, “Space-Air-Ground Integrated Networks: Outage Performance Analysis,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 12, pp. 7897–7912, 2020.

[34] S. Asmussen, *Applied Probability and Queues*. New York, USA: Springer-Verlag, 2003.

[35] G. Zhou, L. Zhao, G. Zheng, Z. Xie, S. Song, and K.-C. Chen, “Joint Multi-Objective Optimization for Radio Access Network Slicing Using Multi-Agent Deep Reinforcement Learning,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 9, pp. 11 828–11 843, 2023.