Satellite Edge Computing for Mobile Multimedia Communications: A Multi-agent Federated Reinforcement Learning Approach

WEIWEI JIANG and YAFENG ZHAN, Tsinghua University, China
XIN FANG, State Grid Jiangsu Electric Power Co., LTD. Research Institute, China

The rapid expansion of satellite mega-constellations has highlighted the potential of satellite edge computing as a promising solution for mobile multimedia communications. While reinforcement learning has been explored in satellite communication systems, significant challenges remain, including high latency and limited resources. This study addresses these challenges by focusing on the joint optimization of communication, computing, and caching resources in satellite edge computing to support mobile multimedia applications. A mixed-integer nonlinear programming (MINLP) problem is formulated with the objective of minimizing the total delay experienced by mobile users, subject to multidimensional resource capacity constraints, which is NP-hard and computationally intractable to solve in polynomial time. To address this complexity, we propose a multi-agent federated reinforcement learning (MAFRL) approach as an efficient solution. In this framework, each satellite operates as an autonomous learning agent equipped with an actor-critic network structure. The proposed MAFRL method demonstrates superior performance, achieving lower delays compared to all baseline approaches. It effectively optimizes delay-sensitive mobile multimedia communications by minimizing total delay and improving task offloading ratios. To the best of the authors' knowledge, this study is the first to introduce a MAFRL-based approach for resource allocation in satellite edge computing, marking a significant contribution to the field.

CCS Concepts: • **Networks → Network management**; • **Computing methodologies → Multi-agent systems**.

Additional Key Words and Phrases: Satellite Edge Computing, Mobile Multimedia Communication, Multi-agent Reinforcement Learning

1 INTRODUCTION

Satellite communication has historically played a significant role in broadcasting, facilitating the widespread distribution of television signals, radio broadcasts, and other media content. However, in the emerging B5G/6G era, satellite communication is expected to assume a more integral role in mobile multimedia communications, characterized by several key advancements. Historically, satellite broadcasting systems were constrained by high latency due to the significant distance between satellites and ground stations, making them less suitable for interactive applications requiring real-time responsiveness. Future B5G/6G satellite systems aim to address these limitations by leveraging advancements in satellite technology, signal processing, and network architecture to achieve low-latency communication. These improvements will support multimedia applications such as virtual reality (VR), augmented reality (AR), and telepresence. Moreover, B5G/6G satellite systems are anticipated to integrate closely with terrestrial 5G/6G networks, creating unified heterogeneous network architectures that extend coverage, increase capacity, and enhance reliability for mobile users. This integration will enable seamless handovers between satellite and terrestrial networks, dynamic resource allocation, and coordinated network

Corresponding author: Yafeng Zhan.
Authors' addresses: Weiwei Jiang, jww@bupt.edu.cn; Yafeng Zhan, zhanyf@mail.tsinghua.edu.cn, Tsinghua University, Beijing, China, 100084; Xin Fang, State Grid Jiangsu Electric Power Co., LTD. Research Institute, Nanjing, China, 211103, 15251867073@126.com.

ACM 1556-4703/2025/2-ART
https://doi.org/10.1145/3715146

resource management to optimize performance and user experience across diverse environments and mobility scenarios.

Low Earth orbit (LEO) communication systems have evolved significantly, driven by advancements in satellite technology and the demand for high-speed, low-latency connectivity. Initially composed of a few satellites primarily used for military and scientific purposes, LEO constellations expanded with the emergence of commercial ventures like Iridium and Globalstar in the late 1990s, which provided global voice and data services. Recently, companies such as SpaceX's Starlink, OneWeb, and Amazon's Project Kuiper have launched ambitious LEO satellite projects to deliver broadband internet to underserved regions and address the increasing demand for connectivity [38]. These modern constellations deploy hundreds to thousands of small satellites, utilizing innovations in miniaturization, launch technologies, and autonomous operations to provide high-speed internet with reduced latency. For instance, Starlink offers upload speeds of 5–10 Mbps, download speeds of 25–100 Mbps, and latencies of 25–60 ms for standard users.

Satellite edge computing (SEC) offers a promising solution to the challenges posed by insufficient terrestrial network coverage in remote areas such as oceans, mountains, and deserts [19]. SEC provides network connectivity for large-scale mobile applications, enabling the monitoring and management of natural resources, such as oceans, forests, and minerals. Typical satellite Internet of Things (IoT) applications include maintaining infrastructure like transportation, logistics, oil pipelines, and power grids, as well as monitoring and predicting geological disasters in forests, rivers, and oceans [26]. The architecture of SEC typically comprises three segments: space, ground, and user. The space segment consists of satellite nodes equipped with onboard payloads that provide communication, computing, and caching resources for vertical industry applications. The ground segment includes satellite gateway stations that manage users and allocate resources while offering low-power communication services. The user segment involves various ground terminals accessing satellite services directly or via ground gateways [4]. Deploying computing capabilities at the satellite edge enhances performance, reliability, autonomy, and security while conserving bandwidth and reducing latency, which is critical for real-time applications like remote sensing, disaster response, and autonomous systems.

The integration of SEC with LEO satellite mega-constellations is increasingly seen as a transformative approach for mobile multimedia communications. For instance, Starlink has been deployed to provide real-time multimedia services. Current research focuses on optimizing resource allocation, task offloading strategies, and network management protocols to minimize latency, improve throughput, and enhance user experiences. Key areas of exploration include content caching and prefetching at satellite edge nodes, dynamic allocation of resources based on user demand, and efficient handover mechanisms between satellite and terrestrial networks. Additionally, researchers are addressing challenges related to security, privacy, and energy efficiency to support seamless integration of satellite communication and edge computing technologies [42].

Despite these advancements, several challenges persist in satellite-based mobile multimedia communication, including high latency in communication links, seamless handovers between networks, resource optimization, security and privacy in distributed environments, and energy efficiency for mobile devices. Interoperability, scalability, and regulatory issues further complicate system design and implementation. To address these challenges, advancements in joint optimization of communication, computing, and caching resources are necessary.

With increasing computing capabilities on satellite nodes, multimedia processing tasks can be offloaded to powerful edge servers in LEO satellites, reducing execution latency and energy consumption for mobile devices. For example, satellites can serve as edge servers to project 2D Fields of View (FoV) into 3D FoVs for mobile VR communication. They can also cache multimedia files, alleviating traffic on satellite backhaul links and enhancing QoS for mobile users [39]. Mobile multimedia communications have diverse applications, such as enhancing communication and control in smart grids. Satellite communication is indispensable for transmitting data from remote distribution stations to data centers, enabling robust data links for monitoring and managing power networks in regions lacking traditional communication infrastructure. Recent advancements in SEC focus on enabling onboard data processing, reducing dependency on long-distance transmissions to Earth, and addressing the growing volume of space-generated data. Companies like Unibap and Spiral Blue are pioneering onboard computing solutions that enhance operational efficiency and reduce latency. Such innovations are poised to transform the satellite industry, projected to become a $16 billion market within the next decade.

This study proposes a novel approach to joint optimization of communication, computing, and caching resources in SEC, modeled as a mixed-integer nonlinear problem (MINLP). Using multi-agent federated reinforcement learning (MAFRL), where satellites act as agents, we demonstrate improved delay performance compared to existing methods. Numerical experiments validate the proposed MAFRL approach, outperforming baselines such as greedy local computing, localized DDQN, and multi-agent Q-learning.

The main contributions of this study are summarized as follows:

- Integration of communication, computing, and caching resources in SEC for mobile multimedia communications.
- Introduction of MAFRL for distributed resource allocation in SEC.
- Experimental validation of MAFRL over traditional approaches including greedy local computing, greedy edge computing, localized DDQN, and multi-agent Q-learning approaches.

The remainder of this paper is structured as follows: Section 2 reviews related work. Section 3 presents the system model and problem formulation. Section 4 outlines the proposed MAFRL solution. Section 5 discusses experimental results. Section 6 concludes the study.

2 RELATED WORK

2.1 Integration of Communication, Computing and Caching in Satellite Edge Computing

Integration of communication, computing, and caching has been a hot area of study in terrestrial networks, particularly in the context of edge computing. This approach aims to combine communication, computation, and caching capabilities at the network edge to improve efficiency, performance, and scalability. Terrestrial edge computing operates within the Earth's atmosphere, typically leveraging fixed or mobile infrastructure deployed on the ground or in proximity to users. In contrast, SEC operates in space, where satellites orbit the Earth at various altitudes. The physical environment of space introduces unique challenges related to radiation, vacuum, temperature extremes, and limited access to resources, which must be addressed in SEC architectures. Terrestrial edge computing benefits from relatively stable and high-bandwidth communication links between edge nodes and end users, often facilitated by wired or wireless terrestrial networks. In SEC, communication links between satellites, ground stations, and user terminals are subject to factors such as line-of-sight constraints, atmospheric interference, and satellite handovers, which can impact connectivity, latency, and reliability. SEC offers global coverage and scalability, allowing access to remote or underserved regions where terrestrial infrastructure may be lacking or prohibitively expensive to deploy. However, ensuring seamless integration and coordination among distributed satellite resources poses unique challenges in terms of orchestration, resource allocation, and load balancing across a dynamic and heterogeneous network of satellites.

Research on the integration of communication, computing, and caching in SEC has advanced significantly with the aim of enhancing the efficiency and performance of satellite networks. Studies have focused on developing novel architectures and algorithms to optimize resource allocation, mitigate latency, and improve scalability. Key advancements include the utilization of machine learning techniques for intelligent caching and routing decisions, design of efficient communication protocols tailored to the unique characteristics of satellite networks, and exploration of edge computing paradigms to offload processing tasks closer to end-users. These efforts have the potential to revolutionize satellite communications by enabling faster data delivery, reducing bandwidth consumption, and supporting emerging applications, such as IoT and immersive multimedia services.

To support the connection of massive users and satisfy the management requirements of various applications in satellite communication systems, many studies have been carried out in the literature on research topics about communication resource management and optimization, including spectrum sharing, random access, multi-beam cooperative transmission, and communication resource allocation [6, 21]. Many methods and techniques have been proposed to improve the efficiency of satellite communication systems, including artificial intelligence, game theory, heuristic algorithms, and convex optimization [2, 5, 29]. A DeepAR-based network slice admission control scheme is proposed for network request prediction and resource allocation in heterogeneous networks, with the aim of increasing resource utilization, slice admission ratio, and revenue of network service operators [20]. To improve the efficiency of limited resources in satellite-terrestrial integrated networks, a multi-sided ascending-price auction mechanism is proposed for slice admission control and resource allocation [18]. Multiple strategic service providers maximize their own utilities by trading bandwidth resources based on the proposed auction mechanism, which has proven to be strongly budget-balanced, individually rational, and obviously truthful. The proposed game theory solution is asymptotically optimal with an optimal mechanism in terms of the admission ratio and service provider profit.

Existing research on satellite communication resource optimization is mostly based on the assumption of a satellite transparent forwarding working mode and fails to consider the joint optimization of on-board computing and caching resources. A satellite transparent forwarding working mode refers to a method of communication where a satellite acts as a transparent intermediary for data transmission between two or more ground-based communication terminals without actively altering the content of the transmitted data. In this mode, the satellite functions as a passive relay, simply receiving signals from one terminal, amplifying or repeating them, and then transmitting them to the intended recipient terminal(s) without modifying the data payload. With the gradual increase in satellite payload capabilities, satellite edge-computing technologies have been proposed to further improve the service capabilities of satellite communication systems.

By deploying edge computing servers on satellites, the task processing delay of mobile multimedia communications can be reduced, and all local data can be avoided from being sent back to the cloud for processing, thereby reducing the data transmission delay. Current satellite edge-computing research is still in the preliminary research stage. With the development and improvement of software-defined satellite technology, network virtualization technology, and other related technologies, there is room for further improvement in the application of SEC to mobile multimedia communications [15].

Satellite edge caching technology can further improve the transmission efficiency of satellite multimedia data collection and distribution services and reduce repeated and unnecessary content transmission, thereby reducing the transmission delay of related services. However, research on satellite edge caching still faces challenges, such as huge data volume, limited on-board storage space, and large satellite link delays. It is necessary to design a reasonable and efficient collaborative transmission and content caching mechanism.

A satellite cache content prediction model is proposed in [34] that can infer the probability of a certain content being cached based on the historical popularity information of the content, and accordingly designed a content-aware routing scheme that maximizes revenue and reduces delay and content retrieval. A caching strategy combining satellite node classification and popular content awareness is proposed in [40]. First, satellite nodes are divided into core and edge nodes based on changes in the connection relationships and interaction sequences during the spatio-temporal evolution of satellite nodes. The probabilistic caching scheme uses core nodes as cache nodes to ensure cache performance, thereby promoting the diversity of the cached content and reducing user request delays and content acquisition hops. Existing research on satellite edge caching technology mainly implements caching configurations in a single dimension, such as content popularity or data freshness, which cannot adapt to the characteristics of massive and polymorphic multimedia data, leading to problems such as low caching efficiency and inflexible business configurations.

With the growth of on-board resources and the integration of various businesses, integrated research ideas have gradually been applied to the joint optimization of on-board resources, including communication, computing, caching, and navigation. For example, an environmental sensing model based on low-orbit satellite downlink signals is proposed in [17], in which a fine-grained three-dimensional rainfall field reconstruction method based on a compressed sensing algorithm is designed and verified using simulated satellite signals and rainfall field distribution data. However, the integration of communication, computing, and caching has only been studied in terrestrial network scenarios, and has not been fully considered in SEC-based mobile multimedia communication scenarios.

In summary, the existing research has mainly focused on the optimization methods of one-dimensional resources in satellite communication systems. There has been insufficient attention and research on the integration of communication, computing, and caching, making it difficult to achieve a significant breakthrough. To meet the growing and increasingly complex mobile multimedia application requirements, it is necessary to study the joint optimization of satellite communication, computing, and caching resources and further improve the comprehensive utilization efficiency of various satellite network resources. Existing studies for the integration of communication, computing and caching in SEC are summarized in Table 1.

Reference	Summary
[20]	A DeepAR-based network slice admission control scheme is proposed for network request prediction and resource allocation.
[18]	A multi-sided ascending-price auction mechanism is proposed for slice admission control and resource allocation.
[34]	A satellite cache content prediction model is proposed.
[40]	A caching strategy combining satellite node classification and popular content awareness is proposed.

Table 1. A summary of existing studies for the integration of communication, computing and caching in SEC.

2.2 Multi-agent Deep Reinforcement Learning in Satellite Edge Computing

Research on deep reinforcement learning (DRL) for SEC has made significant strides, leveraging the power of artificial intelligence to optimize resource management and task scheduling in satellite networks [8]. Studies have focused on developing DRL-based algorithms that are capable of learning complex decision-making policies that adapt to dynamic network conditions and user demands [23]. Advancements include the integration of DRL with edge computing architectures to enable autonomous resource allocation, real-time optimization, and efficient task offloading [7]. These developments hold promise for enhancing the performance, scalability, and reliability of SEC systems, ultimately enabling the delivery of diverse services with improved quality and reduced operational costs. The capacity management problem in a three-layer heterogeneous satellite network is solved using Q-learning, with the optimization objective of maximizing long-term system utilization and reducing storage and computing complexity [14]. DRL is further applied in hybrid satellite and high-altitude platform (HAP) networks to optimize satellite association and HAP location, with the objective of maximizing the end-to-end data rate [22].

Multi-agent deep reinforcement learning (MADRL) deals with scenarios where multiple autonomous agents interact with each other and their environment, aiming to achieve individual or collective goals. It is widely used in autonomous and adaptive systems (AASs), which are systems designed to operate without direct human intervention, adjusting their behavior and responses based on their environment and experience [36]. AAS can employ MADRL techniques to enable autonomous systems to interact with other agents and learn optimal behaviors in complex, dynamic environments. MADRL provides a framework for modeling the interactions between autonomous agents, allowing them to learn from each other's actions and adapt their strategies accordingly. By incorporating MADRL into AAS, systems can exhibit adaptive behaviors not only in response to environmental changes but also in interaction with other autonomous entities [30].

Research on MADRL for SEC has witnessed significant advancements, focusing on developing collaborative decision-making strategies among multiple agents to optimize resource allocation and task scheduling in satellite networks. In normal DRL, there is typically only one agent interacting with the environment. This agent learns to optimize its behavior by receiving feedback from the environment in the form of rewards. In MADRL, there are multiple agents interacting with each other and the environment. These agents may have competing or cooperative objectives, and their actions may affect not only their own rewards but also those of other agents. The interactions between agents introduce additional complexity to the learning process. In multi-agent reinforcement learning (MARL), agents may need to communicate or coordinate with each other to achieve their objectives efficiently. This introduces additional challenges, such as learning effective communication protocols or coordinating actions in real-time.

Studies have explored various approaches to address the challenges of distributed decision making and coordination among agents in dynamic and resource-constrained environments [24]. Key progress includes the design of MADRL frameworks capable of learning communication and collaboration policies among agents, enabling efficient resource utilization, load balancing, and fault tolerance in SEC systems [11]. These advancements hold promise for enhancing the scalability, flexibility, and resilience of satellite networks and paving the way for the deployment of advanced applications and services with improved performance and reliability [32].

An energy-efficient collaborative offloading scheme for SEC is proposed in [43], addressing the constraints of limited energy in LEO satellites and the need to handle diverse computational tasks with varying delay requirements. The proposed service architecture integrates ground edge, satellite edge, and cloud computing platforms, employing an access threshold strategy to balance traffic loads across these platforms. Tasks are categorized into delay-sensitive (DS) and delay-tolerant (DT) types, with DS tasks offloaded to LEO satellites and DT tasks directed to cloud or ground edges to alleviate satellite workload. Performance evaluation using a continuous-time Markov chain model demonstrates substantial energy savings and delay reductions compared to homogeneous task offloading approaches.

To minimize total network delay for terrestrial users in SEC networks, a distributed optimization model leveraging the alternating direction method of multipliers (ADMM) is introduced in [33]. This model collaboratively optimizes the allocation of computational, communication, and storage resources among multiple satellites. By reformulating the non-convex service deployment problem into a convex problem, the model enables efficient polynomial-time solutions. Simulations utilizing global population data indicate that the distributed optimization model significantly improves task completion delay and load balancing, achieving up to 90% performance gains over baseline methods.

In [13], a deep MARL algorithm, MATORA, is proposed for task offloading and resource allocation in SEC environments. The algorithm aims to minimize weighted latency by addressing the dynamic nature of communication channels, queue delays, and varying satellite loads. The problem is decoupled into two sub-problems: task offloading and resource allocation. Task offloading decisions are made using a distributed multi-agent deep reinforcement learning approach, enabling agents to operate independently without requiring prior knowledge of other agents' states. Resource allocation is subsequently handled using convex optimization after fixing the offloading decisions. Extensive simulations demonstrate MATORA's effectiveness in adapting to dynamic conditions, significantly reducing task latency and drop rates.

Dynamic computation offloading in SEC under limited and variable energy supplies in LEO satellites is addressed in [3]. The authors propose a dynamic offloading strategy (DOS) based on Lyapunov optimization theory, aiming to minimize overall task completion time while respecting long-term energy constraints. The DOS employs a hierarchical decomposition method to jointly optimize task offloading, computing, and communication resource allocation. Simulation results show that DOS achieves near-optimal performance and significantly outperforms baseline approaches in terms of task completion time and drop rates, providing a robust solution for energy-constrained SEC scenarios.

To address resource inefficiencies caused by static allocation, two computation offloading strategies for LEO SEC systems are proposed in [35]. The OFDMA-based joint optimization framework for offloading decisions and dynamic resource allocation (ODDRA) dynamically allocates computing and bandwidth resources based on real-time task loads. The TDMA-based joint optimization for offloading decisions and task sequencing (ODTOS) models task offloading sequences as a permutation flow shop problem, solving it with the heuristic Liu and Reeves algorithm. Offloading decisions are managed using matching theory and coalition game theory. Simulation results confirm that both strategies achieve significant reductions in system delay and energy consumption compared to existing methods.

Multi-agent deep reinforcement learning is often combined with game theory to obtain a joint solution [16]. A cooperative multi-agent deep reinforcement learning method is proposed for direct-to-user satellite mobile services, with the optimization objective of bandwidth allocation in which each beam acts as a player in a game-theoretic-based model [12]. A cooperative multi-agent deep reinforcement learning framework is proposed to make decisions on bandwidth allocation and beam patterns in beam hopping satellite systems, with the optimization objectives of maximizing throughput and minimizing delay fairness [25]. Multi-agent reinforcement learning is also used to design a satellite handover strategy, with the optimization objective of handover minimization, subject to the load constraint of each satellite [10]. It is further demonstrated that the multi-agent reinforcement learning scheme outputs a single-agent version in dynamic resource management in very high throughput satellite systems [31]. Existing studies for multi-agent deep reinforcement learning in SEC are summarized in Table 2.

Reference	Summary
[24]	A cooperative multi-agent deep reinforcement learning framework is proposed for radio resource management.
[11]	A multi-agent actor-critic method with attention mechanism is proposed for resource allocation.
[32]	A hybrid satellites network traffic control paradigm with a multi-agent actor-critic algorithm is proposed.
[12]	A cooperative multi-agent deep reinforcement learning method is proposed for direct-to-user satellite mobile services.
[25]	A cooperative multi-agent deep reinforcement learning framework is proposed for bandwidth allocation.
[10]	A satellite handover strategy based on multi-agent reinforcement learning is proposed.

Table 2. A summary of existing studies for multi-agent deep reinforcement learning in SEC.

2.3 Research Gaps and Proposed Solution

Although multi-agent reinforcement learning has been considered in satellite communication systems, research gaps remain. The application of multi-agent reinforcement learning in satellite-edge computing-based mobile multimedia communication scenarios has not been fully considered. A potential solution based on multi-agent reinforcement learning for the specific joint optimization problem of communication, computing, and caching resource allocation is also missing in the literature.

Multi-agent federated reinforcement learning is chosen as the solution in this study because it offers several advantages over traditional reinforcement learning methodologies. Firstly, it enables distributed learning across multiple agents, allowing for more efficient utilization of computational resources and scalability to larger problem domains. Additionally, by decentralizing the learning process, multi-agent federated reinforcement learning can accommodate heterogeneous data sources and diverse learning objectives among the agents, fostering greater flexibility and adaptability in complex environments. Moreover, this approach promotes privacy preservation and data security by allowing agents to learn from local data while sharing only model updates or aggregated information, thus mitigating concerns associated with centralized data storage and processing. Overall, multi-agent federated reinforcement learning represents a promising paradigm for addressing the challenges of scalability, heterogeneity, and privacy in large-scale decentralized systems.

The MAFRL approach introduced in this study represents a significant advancement in the state-of-the-art for SEC in mobile multimedia communications by addressing key limitations of existing solutions. Unlike traditional methods that focus on optimizing individual resources, MAFRL jointly optimizes communication, computing, and caching resources, ensuring a holistic approach to resource allocation. The use of multi-agent reinforcement learning, where each satellite acts as an autonomous learning agent, enables distributed decision-making, improving scalability and reducing computational bottlenecks. Federated learning is employed to facilitate collaboration among satellites, enhancing learning efficiency while preserving data privacy and reducing inter-satellite communication overhead. Additionally, the actor-critic framework with adaptive reward mechanisms ensures more effective handling of complex and dynamic network conditions, significantly lowering total delay and improving task offloading ratios compared to baseline approaches. This integration of advanced reinforcement learning techniques and federated frameworks marks a novel contribution to optimizing delay-sensitive and resource-intensive multimedia applications in SEC environments.

3 SYSTEM MODEL

SEC for multimedia communication offers significant potential in scenarios where terrestrial networks are unavailable, unreliable, or insufficient. One prominent use case is providing high-quality streaming services and real-time communication in remote or underserved regions, such as mountainous areas, oceans, or rural communities, where conventional infrastructure is costly or impractical to deploy. Another scenario is supporting mobile users on vehicles like ships, planes, and trains, ensuring uninterrupted multimedia access during transit. Disaster response is another critical application, where SEC can enable real-time video communication and data sharing among emergency responders in areas affected by natural disasters where ground networks may be damaged. Additionally, this approach is valuable for immersive multimedia applications, such as VR and AR, for industries like remote education, telemedicine, or collaborative engineering, especially in locations lacking robust connectivity. SEC can also enhance content delivery by caching popular media at the edge, reducing latency and bandwidth consumption for global users accessing multimedia-rich applications.

In Fig. 1, there are M mobile users and N satellites in SEC for mobile multimedia communications [9]. As discussed in the Introduction section, the considered satellite type is LEO satellite in this study. In addition to the communication resources, each satellite is equipped with an edge server and an on-board storage database. For multimedia transmission task $k \in \{1, 2, \ldots, K\}$, the required file size is a_k and b_k, where a_k is the local data generated for task k and b_k is the cached file size. The computing requirement ε_k for task k is modeled as the number of CPU cycles. In each time slot, the satellite network topology is assumed to be static snapshots and can be predicted when the satellite orbital elements are known in advance. Each user has at most one multimedia transmission task, e.g., decoding and watching a video clip. The relationship between users and tasks is modelled as a binary variable $\xi_{jk} \in \{0, 1\}$, $k \in \{1, 2, \ldots, K\}$, $j \in \{1, 2, \ldots, M\}$, where $\xi_{jk} = 1$ indicates that task k is performed by user j.

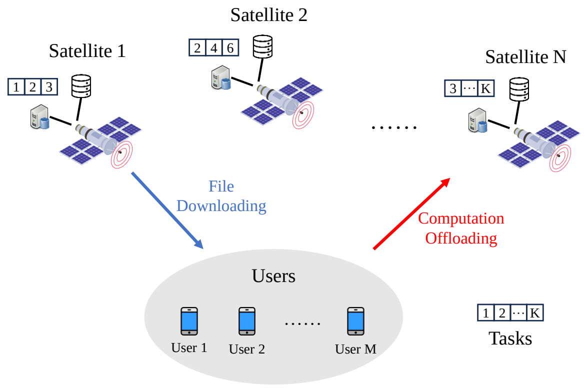

Fig. 1. System model of SEC for mobile multimedia communications.

Two binary decision variables α_{ij} and β_{ij} are introduced to model different resource allocation schemes, where $\alpha_{ij} \in \{0, 1\}$ represents whether the task from user j is computed in the edge server from satellite i (i.e., $\alpha_{ij} = 1$ if it is the case), and $\beta_{ij} \in \{0, 1\}$ represents whether the multimedia file requested by the task of user j is downloaded from satellite i.

In this section, we first derive the total delay and resource constraints in the local computing, edge computing, and edge caching modes. We then formulate the joint optimization problem of communication, computing, and caching resource allocation.

3.1 Local Computing

In the local computing mode, the multimedia file requested by user j is downloaded from satellite i. The downlink rate R_{ij}^D (bits/s) between satellite i and user j is formulated as follows:

$$R_{ij}^D = y_{ij} \cdot B^D \cdot \log_2(1 + \frac{P_{ij}^D \cdot g_{ij}^D}{\sigma^2}) \tag{1}$$

where y_{ij} is the bandwidth allocation ratio of satellite i for user j, B^D is the downlink bandwidth, P_{ij}^D is the transmission power from satellite i to user j, g_{ij}^D is the channel gain from satellite i to user j, and σ^2 is the noise power. In this study, a homogeneous satellite network is used in which each satellite has the same downlink bandwidth.

The total delay in the local computing mode consists of three parts: the file transmission time, task execution time, and propagation delay. The total delay of user j in the local computing is calculated as follows:

$$T_{ij}^L = \sum_{i=1}^{N} \beta_{ij} \left(\frac{\sum_{k=1}^{K} \xi_{jk} \cdot b_k}{R_{ij}^D} + \frac{\sum_{k=1}^{K} \xi_{jk} \cdot \varepsilon_k}{f_j} + t_{ij}^{\text{prop}} \right) \tag{2}$$

where f_j is the local computing capacity of user j, t_{ij}^{prop} is the one-way propagation delay between satellite i and user j.

The energy consumption of user j in local computing is calculated as follows.

$$E_j^L = \sum_{i=1}^{N} \beta_{ij} \cdot \frac{\sum_{k=1}^{K} \xi_{jk} \cdot \varepsilon_k}{f_j} \cdot P_j^L \tag{3}$$

where P_j^L is the power consumption of CPU for user j. P_j^L can be calculated as

$$P_j^L = \kappa \cdot f_j^3 \tag{4}$$

where κ is the energy consumption coefficient [37]. By using $P_j^L = \kappa \cdot f_j^3$, the energy consumption E_j^L becomes

$$E_j^L = \sum_{i=1}^{N} \beta_{ij} \cdot \sum_{k=1}^{K} \xi_{jk} \cdot \varepsilon_k \cdot \kappa \cdot f_j^2 \tag{5}$$

3.2 Edge Computing

In edge computing mode, the generated local data are offloaded to a satellite node for further processing. Similar to the data download process, the upload data rate between satellite i and user j is calculated as:

$$R_{ij}^U = x_{ij} \cdot B^U \cdot \log_2(1 + \frac{P_{ij}^U \cdot g_{ij}^U}{\sigma^2}) \tag{6}$$

where x_{ij} is the uplink communication bandwidth ratio allocated from satellite i and user j, B^U is the total uplink bandwidth, P_{ij}^U is the uplink transmission power from user j to satellite i, g_{ij}^U is the channel gain from user j to satellite i. Similar to existing studies [41], the task execution result download time is negligible. However, both the upload and download propagation delays are considered when calculating the total delay.

The total delay in the edge computing mode is calculated as:

$$T_j^E = \sum_{i=1}^{N} \alpha_{ij} \left(\frac{\sum_{k=1}^{K} \xi_{jk} \cdot a_k}{R_{ij}^U} + \frac{\sum_{k=1}^{K} \xi_{jk} \cdot \varepsilon_k}{z_{ij} \cdot F_i^S} + 2t_{ij}^{\text{prop}} \right) \tag{7}$$

where z_{ij} represents the computing resource allocation ratio from satellite i to user j, F_i^S is the computing capacity of satellite i.

The energy consumption of user j in edge computing mode is calculated as

$$E_j^E = \sum_{i=1}^{N} \alpha_{ij} \cdot \frac{P_j^U \cdot \sum_{k=1}^{K} \xi_{jk} \cdot a_k}{R_{ij}^U} \tag{8}$$

where P_j^U is the power consumption for task offloading of user j.

The successful operations of both local and edge computing rely on the edge caching of the required multimedia files in satellite edge nodes. To describe this constraint, a binary variable $s_{ik} \in \{0, 1\}$ is introduced, and $s_{ik} = 1$ if the required file of task k is cached to satellite i. The satellite edge caching resource availability constraint is represented as

$$(\alpha_{ij} + \beta_{ij}) \cdot \xi_{jk} \le s_{ik}, \forall i, j, k \tag{9}$$

Each satellite node has limited storage capacity C_i^S for data caching. The satellite edge caching capacity constraint is represented as

$$\sum_{k=1}^{K} s_{ik} \cdot b_k + \sum_{j=1}^{M} \alpha_{ij} \sum_{k=1}^{K} \xi_{jk} a_k \le C_i^S, \forall i \tag{10}$$

3.3 Optimization Problem Formulation

Considering that delay is the main concern in mobile multimedia transmission, the optimization objective is the total delay. In our future research, we would also take other network metrics, e.g., transmission rate and network reliability, into the optimization problem formulation. The optimization problem is formulated as follows:

$$
\begin{aligned}
&min_{\alpha,\beta,x,y,z,s} \sum_{j=1}^{M}(T_j^L + T_j^E) \\
&s.t.\ \text{C1: } E_j^L + E_j^E \leq E_j^U, \forall j \\
&\quad \text{C2: } \sum_{j=1}^{M} \alpha_{ij} \cdot z_{ij} \leq 1, \forall i \\
&\quad \text{C3: } \sum_{k=1}^{K} s_{ik} b_k + \sum_{j=1}^{M}(\alpha_{ij} \sum_{k=1}^{K} \xi_{jk} a_k) \leq C_i^S, \forall i \\
&\quad \text{C4: } \sum_{i=1}^{N}(\alpha_{ij} + \beta_{ij}) \leq 1, \forall j \\
&\quad \text{C5: } (\alpha_{ij} + \beta_{ij})\xi_{jk} \leq s_{ik}, \forall i, j, k \\
&\quad \text{C6: } \sum_{j=1}^{M} \alpha_{ij} x_{ij} \leq 1, \forall i \\
&\quad \text{C7: } \sum_{j=1}^{M} \beta_{ij} y_{ij} \leq 1, \forall i \\
&\quad \text{C8: } \alpha_{ij}, \beta_{ij}, s_{ik} \in \{0, 1\}, \forall i, j, k \\
&\quad \text{C9: } x_{ij}, y_{ij}, z_{ij} \in [0, 1], \forall i, j
\end{aligned}
\tag{11}
$$

where C1 is the energy consumption constraint for users in which E_j^U is the maximum energy of user j, C2 is the edge computing capacity constraint for satellites, C3 is the edge caching capacity constraint for satellites in which C_i^S is the maximum storage capacity of satellite i, C4 requires that each user can only be associated with one and only one satellite, C5 is the satellite edge caching resource availability constraint, C6 is the satellite upload bandwidth allocation ratio sum constraint, C7 is the satellite download bandwidth allocation ratio sum constraint, C8 requires that α_{ij}, β_{ij}, and s_{ik} are binary variables, C9 requires that x_{ij}, y_{ij}, and z_{ij} are within the range between 0 and 1. The formulated problem is MINLP, which is a typical NP-hard problem that cannot be efficiently solved in polynomial time [1].

4 MULTI-AGENT FEDERATED REINFORCEMENT LEARNING SOLUTION

In this study, we conceptualize multidimensional resource allocation as a dynamic Markov process (MDP). In each time slot, the network state changes based on resource allocation decisions, and the reward is defined. Considering the increase in both satellites and users in LEO mega-constellations and NP-hardness of the formulated optimization problem, it becomes unrealistic to obtain a desirable solution within an acceptable time. Instead, real-time decisions are required for real-world mobile multimedia communication systems.

Reinforcement learning is a promising solution for addressing such challenges. Each satellite node can serve as a learning agent and choose the best resource allocation decisions. Considering the limited computing and learning capacities of mobile users, they follow the instructions provided by satellites. Furthermore, satellites can collaborate and improve learning performance in a federated learning approach, for example, through inter-satellite links.

4.1 The Markov Decision Process

The basic components of an MDP include state, action, and reward, which are defined as follows:

State: State s_i at satellite edge node i contains local observations of the environment, including the communication, computing, and caching states, and the task requests.

Action: In each time slot, action at satellite i is to select the resource allocation and file decisions $a_i = \{\alpha_{ij}, \beta_{ij}, s_{ik}, x_{ij}, y_{ij}, z_{ij}\}, \forall j, k$. While $\alpha_{ij}, \beta_{ij}, s_{ik}$ are binary variables, x_{ij}, y_{ij}, z_{ij} are continuous variables. To narrow down the action search space, the discrete value space for resource allocation ratios are used, i.e., $x_{ij}, y_{ij}, z_{ij} \in \{0, 0.1, \ldots, 1\}$.

Reward: The reward for an agent serves as the key signal used to guide the learning process towards achieving the optimization objective. Because the objective of the formulated optimization problem is to reduce the total delay, the reward r_i is designed as the sum of the negative delay and penalty of constraints:

$$\begin{aligned} r_i = &- \sum_{j=1}^{M} \beta_{ij} \left(\frac{\sum_{k=1}^{K} \xi_{jk} \cdot b_k}{R_{ij}^D} + \frac{\sum_{k=1}^{K} \xi_{jk} \cdot \varepsilon_k}{f_j} + t_{ij}^{\text{prop}} \right) - \sum_{j=1}^{M} \alpha_{ij} \left(\frac{\sum_{k=1}^{K} \xi_{jk} \cdot a_k}{R_{ij}^U} + \frac{\sum_{k=1}^{K} \xi_{jk} \cdot \varepsilon_k}{z_{ij} \cdot F_i^S} + 2t_{ij}^{\text{prop}} \right) \\ &+ \lambda \cdot [\max(0, E_j^L + E_j^E - E_j^U)]^2 + \lambda \cdot [\max(0, \sum_{j=1}^{M} \alpha_{ij} \cdot z_{ij} - 1)]^2 \\ &+ \lambda \cdot [\max(0, \sum_{k=1}^{K} s_{ik} b_k + \sum_{j=1}^{M} (\alpha_{ij} \sum_{k=1}^{K} \xi_{jk} a_k) - C_i^S)]^2 \\ &+ \lambda \cdot [\max(0, \alpha_{ij} + \beta_{ij} - 1)]^2 + \lambda \cdot [\max(0, (\alpha_{ij} + \beta_{ij}) \xi_{jk} - s_{ik})]^2 \\ &+ \lambda \cdot [\max(0, \sum_{j=1}^{M} \alpha_{ij} x_{ij} - 1)]^2 + \lambda \cdot [\max(0, \sum_{j=1}^{M} \beta_{ij} y_{ij} - 1)]^2 \end{aligned} \tag{12}$$

where λ is a penalty parameter. As long as λ is sufficiently large, no constraint violations are considered.

4.2 Multi-agent Actor-Critic Framework

Given the definitions above, the basic structure of the multi-agent reinforcement learning solution is shown in Fig. 2.

In this study, a multi-agent actor-critic framework is used, with four neural networks in each agent, including current and target actor networks, and current and target critic networks [27]. The current and target actor networks have the same feed-forward neural network structure, as shown in Fig. 3. The purpose of the actor network is to determine the action output that maximizes the long-term reward (known as the Q-value in reinforcement learning) based on the state input.

The current and target critic networks also have the same feed-forward neural network structure, as shown in Fig. 4. There are two significant differences between the critic and actor networks. The critic network takes actions as an additional input, compared with the actor network. The output of the critic network is the reward, while the output of the actor network is the action.

The purpose of the critic network is to approximate the reward value output based on state and action inputs. Target networks with different update frequencies are introduced to prevent the training process from spiraling around and improve the stability and convergence of the reply training. In contrast to current networks, target networks estimate future action a_i' and Q value $Q'(s_i', a_i')$ for the next time slot. The use of current and

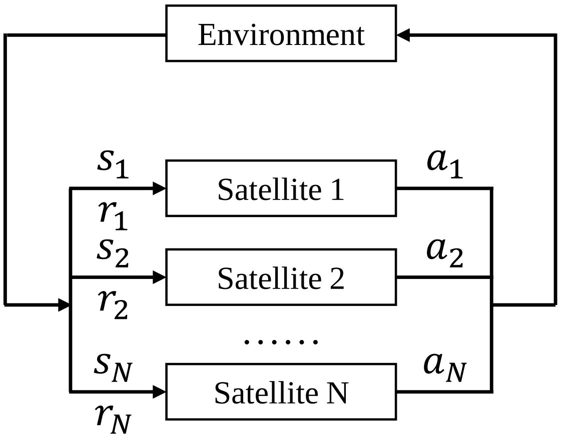

Fig. 2. The basic structure of the multi-agent reinforcement learning solution.

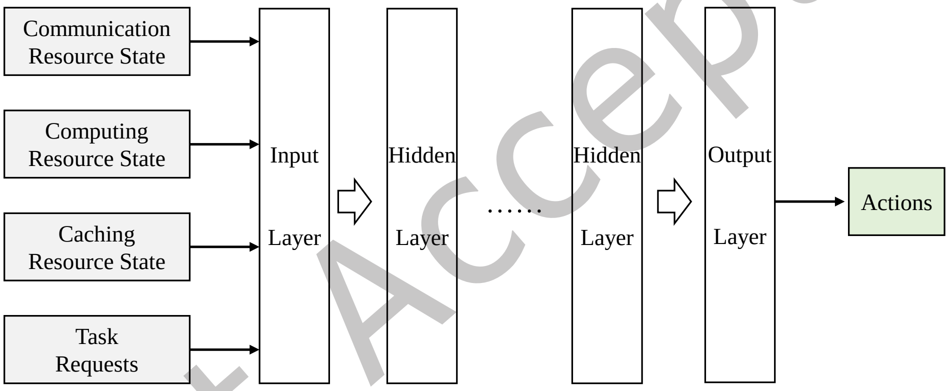

Fig. 3. The actor network structure.

target networks in MAFRL is essential for stabilizing the training process, addressing non-stationarity, reducing correlation between experiences, and improving convergence properties. Training DRL agents, especially in multi-agent environments, can be highly unstable due to the non-stationarity of the environment and the correlation between consecutive experiences. By using separate current and target networks, the learning process is stabilized. This stability is crucial for ensuring that the training process converges to meaningful solutions.

In every T_u time slots, the network parameters of the target actor and critic networks are updated as follows:

$$\theta_i' = \delta\theta_i' + (1-\delta)\theta_i \tag{13}$$

$$\phi_i' = \delta\phi_i' + (1-\delta)\phi_i \tag{14}$$

where θ_i' is the network parameter of the target actor network, θ_i is the network parameter of the current actor network, ϕ_i' is the network parameter of the target critic network, ϕ_i is the network parameter of the current critic network, δ is the soft update weight.

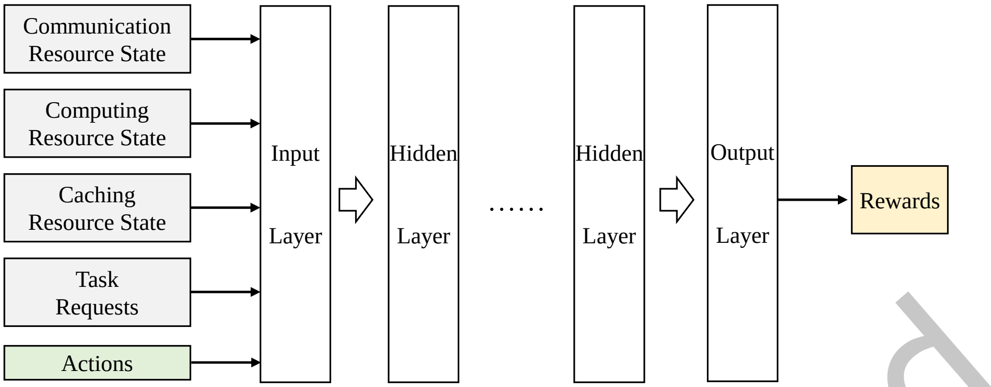

Fig. 4. The critic network structure.

An experience pool is also introduced to enhance the stability and convergence of the deep reinforcement learning model training process. An experience is defined as a tuple (s_i, a_i, r_i, s_i') and is stored in the experience pool $\mathcal{B}$. In each training step, a batch of B samples is obtained from the experience pool.

The loss function of the critic network is the mean squared error:

$$\mathcal{L}_{C_i}(\phi_i) = \sqrt{(C_i(s_i, a_i; \phi_i) - \hat{y}_i)^2} \tag{15}$$

where C_i is the current critic network and $\hat{y}_i$ is the discounted long-term Q value and calculated with the target critic network as:

$$\hat{y} = r_i + \gamma C_i'(s_i', a'; \phi_i') \tag{16}$$

where C_i' is the target critic network.

The loss function of the actor network is the predicted Q value

$$\mathcal{L}_{A_i}(\theta_i) = C_i(s_i, A_i(s_i; \theta_i), \phi_i) \tag{17}$$

where A_i is the current actor network.

Then, the parameters of the current actor and critic networks are updated with stochastic gradient descent (SGD) optimization as follows:

$$\theta_i \leftarrow \theta_i - \eta_A \nabla_\theta \mathcal{L}_{A_i}(\theta_i) \tag{18}$$

$$\phi_i \leftarrow \phi_i - \eta_C \nabla_\phi \mathcal{L}_{C_i}(\phi_i) \tag{19}$$

where η_A and η_C are learning rates for the actor and critic networks.

Finally, federated learning is introduced to enhance the system privacy and training convergence. For every N_u time slots, the FedAvg algorithm [28] is invoked to aggregate the current actor network parameters:

$$\theta \leftarrow \text{FedAvg}(\theta) \tag{20}$$

where $\theta = [\theta_1, \theta_2, \ldots, \theta_N]$ is the parameter vector of all actor networks.

FedAvg represents a promising approach for decentralized model training in SEC environments, offering privacy-preserving, resource-efficient, and effective collaborative learning capabilities. Firstly, FedAvg leverages the distributed nature of edge devices to enable privacy-preserving model training, as data remains local and is not required to be shared centrally. This decentralized approach mitigates privacy concerns associated with data sharing, making FedAvg suitable for applications in sensitive domains such as healthcare and finance. Secondly, FedAvg promotes efficient utilization of computational resources by allowing edge devices to perform local model updates using their own data and computational power. This decentralized training paradigm reduces communication overhead and minimizes the need for large-scale data transfers, making FedAvg well-suited for environments with limited bandwidth or high-latency connections. Moreover, FedAvg employs techniques such as model aggregation and adaptive learning rates to ensure that the global model converges to an accurate representation of the data distributed across edge devices, enabling effective collaborative learning without compromising data privacy or efficiency.

The pseudo code for the multi-agent federated reinforcement learning algorithm is summarized in Algorithm 1. Algorithm 1 outlines the pseudo-code for the MAFRL approach, detailing the steps for training and deployment in a distributed SEC environment. The algorithm begins by initializing the parameters of the actor and critic networks, with separate target networks for stabilizing training. Each satellite agent interacts with the environment over a series of episodes and time steps, either by exploring random actions during initial learning or leveraging its actor network to make informed decisions based on observed states. These interactions generate experiences in the form of state-action-reward-next state tuples, which are stored in an experience pool. At each training step, samples from this pool are used to update the actor and critic networks via stochastic gradient descent, guided by loss functions derived from the predicted and actual Q-values. Periodically, target networks are updated using a soft update rule to enhance stability, and federated learning is employed to aggregate actor network parameters across agents, promoting collaborative learning while preserving local data privacy. This combination of decentralized decision-making, experience replay, and federated learning ensures that the MAFRL approach is both scalable and robust, capable of adapting to dynamic network conditions and diverse resource demands.

5 EXPERIMENTS

5.1 Settings

The experimental setup for evaluating the MAFRL approach is designed to simulate a realistic SEC environment using parameters derived from a typical LEO satellite constellation, such as SpaceX's Starlink. The simulation considers 1,440 satellites in the first stage of Starlink with an orbital altitude of 550 km, distributed across 72 orbital planes with 20 satellites per plane. For the area of interest in a single time slot, 4 satellites are selected as N, and the number of users M varies between 10 and 100, with a default value of 50. The number of tasks K is set to 10, with task data sizes ranging between 0.5 and 5 Mbits and a default value of 2.5 Mbits, where 20%-40% of each task is local data. The upload and download bandwidths (B^U and B^D) are 500 MHz and 2,000 MHz, respectively. SEC capacity F_i^S ranges from 2 to 16 Gcycles/s (default: 10 Gcycles/s), while user local computing capacity f_j ranges from 0.2 to 2 Gcycles/s (default: 1 Gcycles/s). The experiments assume an energy consumption coefficient κ of 10^{-28} and a CPU cycle computing requirement ε_k of 1,000 cycles per bit. The data size of each task $a_k + b_k$ is chosen from [0.5, 5] Mbits and the default value is 2.5 Mbits, where a_k randomly accounts for 20%-40% [9].

For MAFRL, the penalty parameter λ is set to 500, batch size B is set to 128, number of episodes N_e is set to 500, number of time slots T is set to 200, federated learning period N_u is set to 10, and target network update period T_u is set to 10. Two hidden layers with 100 neurons each and ReLU activation are employed for actor and critic networks. These parameters are chosen to balance computational feasibility with the need to model realistic network dynamics. Scenarios tested include varying user densities, task sizes, satellite capacities, and bandwidth settings to evaluate the approach under diverse conditions and assess its adaptability and efficiency in optimizing total delay and task offloading ratios.

Baselines include greedy local computing, greedy edge computing, localized DDQN and multi-agent Q-learning approaches. The greedy local computing approach allows mobile users to use local computing resources as much as possible. Instead, the greedy edge computing enables the usage of satellite edge computing and caching resources as much as possible. The localized DDQN approach employs the neural networks in satellite networks without cooperation. The multi-agent Q-learning method employs the Q-learning algorithm [36]. The evaluation

Algorithm 1 The pseudo-code for the multi-agent federated reinforcement learning algorithm.

```
Initialize current network parameters randomly $\theta, \phi$
Initialize target network parameters $\theta' \leftarrow \theta, \phi' \leftarrow \phi$
for episode = $1, \ldots, N_e$ do
  for step $t = 1, \ldots, T$ do
    for each agent $i \in \{1, 2, \ldots, M\}$ do
      if $|\mathcal{B}_i| < B$ then
        Randomly select actions $a_i$;
      else
        Receive observations $s_i$;
        Obtain actions $a_i = A_i(s_i; \theta_i)$;
      end if
    end for
    Interact with the environment to gather $\{s, a, r, s'\}$, and append it to the experience pool $\mathcal{B}$;
    for each agent $i \in \{1, 2, \ldots, M\}$ do
      if $|\mathcal{B}_i| > B$ then
        Sample $\{s_i, a_i, r_i, s'_i\}$ from $\mathcal{B}_i$;
        Obtain new actions by $a'_i = A'_i(s'_i, \theta'_i)$;
        Obtain new Q values by $Q'(s'_i, a'_i) = C'_i(s'_i, a'_i; \phi'_i)$;
        Calculate discounted long-term Q value by $\hat{y} = r_i + \gamma C'_i(s'_i, a'; \phi'_i)$;
        Calculate loss function of actor and critic networks by $\mathcal{L}_{C_i}(\phi_i) = \sqrt{(C_i(s_i, a_i; \phi_i) - \hat{y}_i)^2}$ and $\mathcal{L}_{A_i}(\theta_i) = C_i(s_i, A_i(s_i; \theta_i), \phi_i)$;
        Update current network parameters using $\theta_i \leftarrow \theta_i - \eta_A \nabla_\theta \mathcal{L}_{A_i}(\theta_i)$ and $\phi_i \leftarrow \phi_i - \eta_C \nabla_\phi \mathcal{L}_{C_i}(\phi_i)$;
      end if
      if $t$ mod $T_u = 0$ then
        Update target network parameters by $\theta'_i = \delta\theta'_i + (1 - \delta)\theta_i$ and $\phi'_i = \delta\phi'_i + (1 - \delta)\phi_i$;
      end if
      if $t$ mod $N_u = 0$ then
        Perform the FedAvg algorithm by $\theta \leftarrow \mathrm{FedAvg}(\theta)$;
      end if
    end for
  end for
end for
```

metrics include total delay and task offloading ratio. The total delay reflects the QoS performance of mobile multimedia services and the task offloading ratio measures the utilization of satellite edge computing and caching resources. Task offloading ratio is not often seen in the literature as an evaluation metric. It is used as the evaluation metric in this study because task offloading ratio measures the proportion of tasks offloaded from the user to the SEC infrastructure. Maximizing this ratio ensures efficient utilization of available resources, including satellite bandwidth and edge computing capacity. Offloading tasks to the edge can significantly reduce end-to-end latency compared to processing them solely on the user device. More evaluation metrics would be considered in our future studies, e.g., handover latency and throughput metrics. The models and algorithms are implemented with Python and TensorFlow 2.0. The experiments are conducted on a Windows desktop computer with 16 GB RAM and 8GB GPU.

5.2 Results and Discussion

In the experiments, we focus on the performance comparison between the proposed MAFRL approach with baselines, in terms of average delay per user for a fair comparison, instead of the total delay of all users, which is highly affected by the number of mobile users. The average delay v.s. data size is shown in Fig. 5. The intuition is that as the data size to be processed increases, transmission and computation delays increase, in both local or SEC modes. Fig. 5 demonstrates that the proposed MAFRL approach achieves a lower delay than all baselines. It is also observed that the performance gap between greedy edge computing and greedy local computing becomes smaller with a larger data size, because the advantage of a higher computing capacity in the satellite edge server becomes more significant when the amount of data increases. The result in Fig. 5 indicates that SEC is desirable for processing large multimedia files in mobile communications, which is beyond the processing capacities of mobile devices.

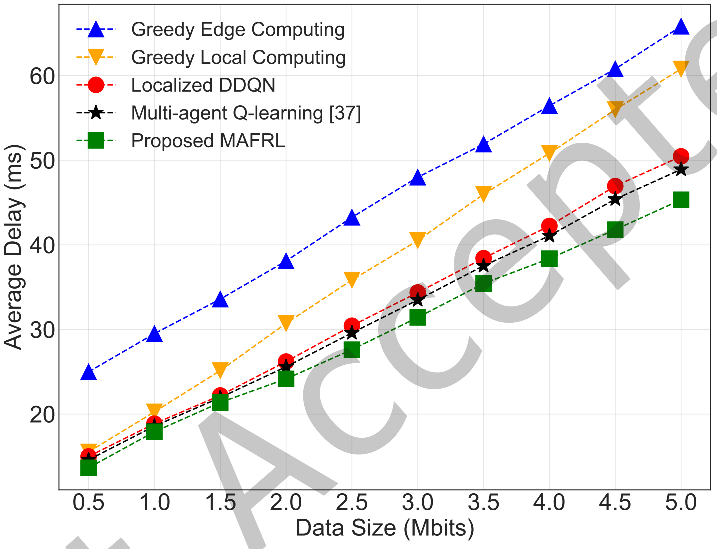

Fig. 5. The average delay v.s. data size.

One of the key enablers of SEC is the growth of satellite computing capacities. In Fig. 6, the average delay v.s. satellite computing capacity for different schemes is evaluated. In the greedy local computing baseline, almost all tasks are performed in the user devices and the average delay is barely affected by the change of satellite computing capacity. The influence of satellite computing capacity on the average delay of the greedy edge computing baseline is more obvious in Fig. 6. However, the influence of satellite computing capacity on the proposed MAFRL and the localized DDQN schemes is more complex. When the satellite computing capacity is minimal, the performance of the proposed MAFRL and the localized DDQN schemes is more close to the greedy local computing scheme, which acts as an upper bound for both solutions and no tasks would be offloaded to the satellites in the extreme case if there is no on-board computing capacity. When the satellite computing capacity is significantly more than the local computing capacity, the performance of the proposed MAFRL and the localized DDQN schemes is more close to the greedy edge computing scheme, when almost all tasks are offloaded to the satellite edge nodes. To further validate these observations, the offloading ratio v.s. satellite computing capacity is further shown in Fig. 7. As expected, the offloading ratio for greedy local computing is almost 0, and the

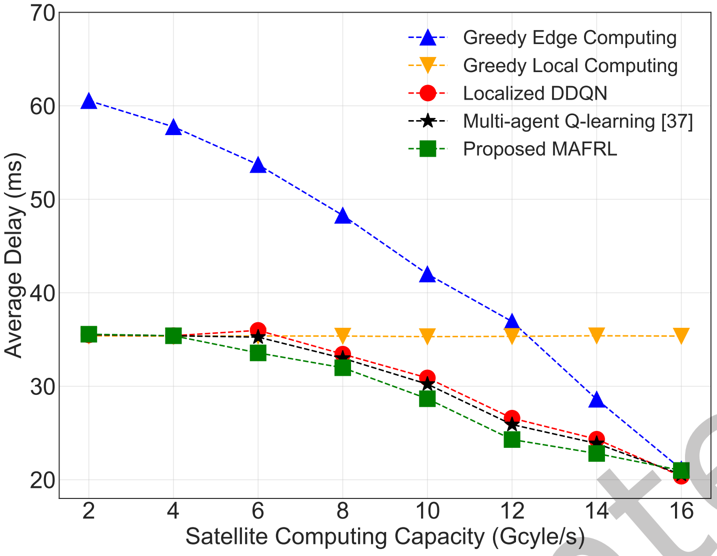

Fig. 6. The average delay v.s. satellite computing capacity.

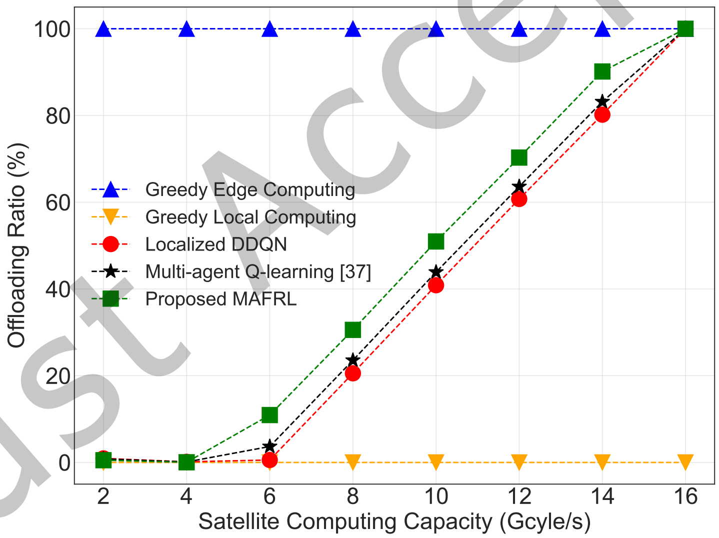

Fig. 7. The offloading ratio v.s. satellite computing capacity.

offloading ratio for greedy edge computing is almost 100%. The offloading ratios for the proposed MAFRL and the localized DDQN schemes both increase with a stronger satellite computing capacity.

The average delay v.s. local computing capacity is shown in Fig. 8. The influence of local computing capacity is contrary to that of SEC capacity. With a stronger local computing capacity, more tasks would be executed locally, as validated in Fig. 9. In Fig. 8, the average delay does not decrease with the local computing capacity for the

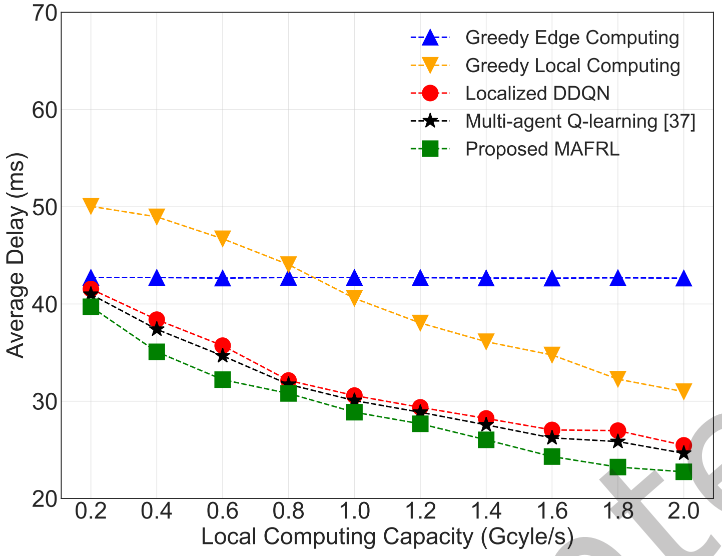

Fig. 8. The average delay v.s. local computing capacity.

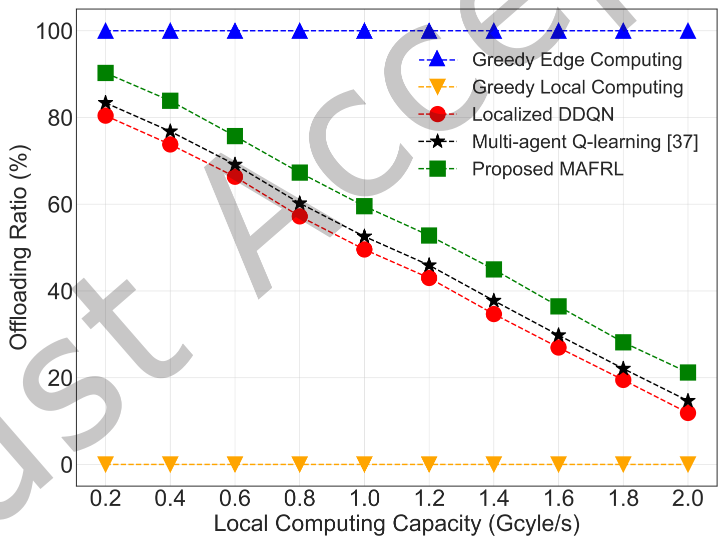

Fig. 9. The offloading ratio v.s. local computing capacity.

greedy edge computing method, because all the tasks are 100% offloaded to the satellite edge server, as validated in Fig. 9. For the remaining methods, the average delay decreases with the increased local computing capacity when part of or all tasks are executed locally.

The influence of the number of mobile users is evaluated in Fig. 10 and Fig. 11, for average delay and offloading ratio. To gain a higher revenue when providing mobile multimedia communication services, a large number of

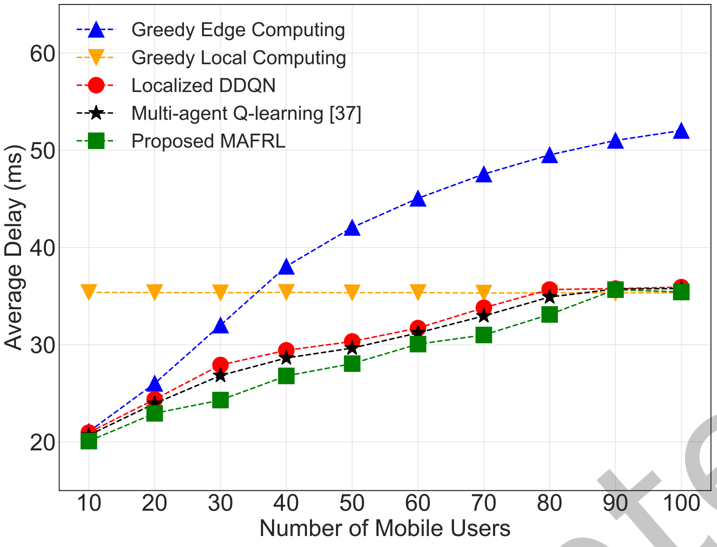

Fig. 10. The average delay v.s. number of users.

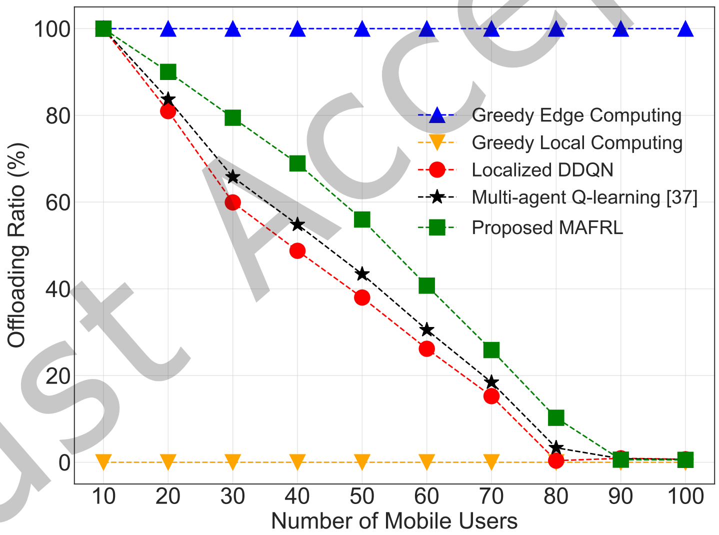

Fig. 11. The offloading ratio v.s. number of users.

users is preferred by the network operator. However, it is also a worry if QoS decreases when the number of users increases. From Fig. 10, the increase of average delay is inevitable when the limited network resources are used to provide services to more users. The average delay for greedy local computing is still not affected by the growth of users because each user is equipped his own capacities and would not affect others. However, the average delay for greedy edge computing is highly affected by the growth of users when the satellite communication, computing, and caching resources becomes insufficient. It is also observed that the proposed MAFRL and the localized DDQN schemes are less affected by the number of mobile users, when they are smart enough to switch between local computing and edge computing edges.

Finally, it is worth mentioning that the proposed federated learning-based MAFRL approach outperforms the localized DDQN baseline, which indicates that the cooperation mechanism based on the FedAvg algorithm is effective when providing mobile multimedia communications with SEC. The results in this study also indicates that it is worth exploring more efficient federated learning mechanisms for SEC.

To conclude this section, some take-home conclusions and observations are summarized as follows:

- MAFRL exhibits lower delay compared to all baselines, showcasing its effectiveness in optimizing delay-sensitive mobile multimedia communications.
- The performance gap between greedy edge computing and greedy local computing diminishes with larger data sizes, favoring SEC due to its higher computing capacity.
- The growth of satellite computing capacities is a key enabler for SEC, where the performance of MAFRL and localized DDQN schemes varies based on satellite computing capacity relative to local computing capacity.
- The offloading ratio increases with stronger satellite computing capacity, with significant offloading observed for MAFRL and localized DDQN schemes.
- The influence of the number of mobile users on delay and offloading ratio underscores the importance of efficient resource allocation, with MAFRL and localized DDQN schemes exhibiting resilience to user growth through smart switching between local and edge computing.
- The federated learning-based MAFRL approach outperforms the localized DDQN baseline, indicating the effectiveness of the FedAvg-based cooperation mechanism for mobile multimedia communications with SEC.

The findings of this study have broader implications for the future of SEC and its role in advancing global connectivity and digital inclusion. By demonstrating the effectiveness of the MAFRL approach in optimizing communication, computing, and caching resources, the study highlights the potential of SEC to support delay-sensitive applications such as real-time video streaming, VR/AR, and telepresence in regions lacking robust terrestrial infrastructure. This capability can bridge the digital divide, providing equitable access to multimedia services for remote and underserved areas. The contributions of this study also extend to enabling more efficient and scalable IoT deployments, such as remote monitoring in agriculture, logistics, and smart grids, by leveraging edge intelligence to reduce latency and enhance reliability. In the larger context, this research aligns with ongoing advancements in satellite mega-constellations and 5G/6G networks, offering a pathway for integrating satellite and terrestrial systems into seamless, heterogeneous networks. Furthermore, the proposed techniques have implications for privacy-preserving AI and distributed learning, addressing growing concerns about data security in decentralized systems. These findings not only set a benchmark for SEC research but also pave the way for innovative applications in global communications, emergency response, and industrial automation.

Implementing the MAFRL approach in real-world satellite systems presents several practical challenges that must be addressed to ensure its viability. One significant issue is the inherent communication delays in satellite networks, particularly for inter-satellite links and between satellites and ground stations, which can impact the timeliness and efficiency of federated learning updates and resource allocation decisions. Hardware limitations, such as constrained computational power, storage capacity, and energy availability on satellites, pose additional challenges for deploying complex reinforcement learning models and executing large-scale collaborative learning processes. Moreover, regulatory constraints related to spectrum allocation, data sharing, and cross-border satellite operations can complicate the deployment of MAFRL-based systems, especially in scenarios requiring cooperation between satellites from different countries or organizations. Addressing these challenges will require advancements in satellite hardware, efficient algorithm design tailored to resource-constrained environments, and international coordination to establish standardized frameworks for regulatory compliance.

The scalability of the MAFRL approach with the number of satellites and users is also a key consideration for its practical deployment in large-scale satellite constellations. The distributed nature of MAFRL, where each satellite acts as a learning agent, inherently supports scaling by allowing satellites to make independent decisions while collaborating via federated learning. This decentralized structure reduces the computational and communication burden on any single satellite, enabling the approach to handle increasing numbers of satellites and users. However, as the network grows, the increased volume of inter-agent communication for federated updates could introduce significant overhead and latency, especially in scenarios with limited inter-satellite link bandwidth. Additionally, the complexity of decision-making rises with larger state and action spaces, potentially requiring more sophisticated neural network architectures and longer training times. Hardware constraints, such as limited onboard processing power and energy, may also hinder scalability, particularly in resource-intensive applications. To address these limitations, future enhancements could include hierarchical learning frameworks, where satellites are grouped into clusters for localized decision-making, or the use of lightweight learning algorithms tailored to constrained environments.

6 CONCLUSION

SEC is a promising solution for mobile multimedia communications owing to its advantages of global coverage and seamless connectivity. The integration of multidimensional resources in space, including communication, computing, and caching, further improves the QoS for mobile users. To jointly optimize resource utilization and decrease the total delay in multimedia transmission tasks, we propose a multi-agent federated reinforcement learning solution in this study, in which each satellite acts as a learning agent and an actor-critic network structure is trained and deployed to make resource allocation decisions in a distributed fashion. The proposed solution is compared with greedy local computing, greedy edge computing, and localized DDQN approaches. Numerical experiments demonstrate that the proposed solution achieves a promising performance with a lower delay, compared with baselines.

Future research directions in SEC for mobile multimedia communications and the MAFRL algorithm can build on the promising results of the proposed solution by exploring various optimization and enhancement avenues. Dynamic resource allocation strategies that adapt to real-time changes in network conditions and user demands, incorporating factors such as weather or satellite positioning, could significantly improve resource utilization and reduce total delay. Integrating heterogeneous satellite networks, including GEO, MEO, and LEO systems, offers the potential to leverage their respective strengths to optimize coverage, latency, and overall QoS. The MAFRL algorithm could be enhanced with adaptive reward mechanisms that consider diverse QoS metrics like energy efficiency and reliability while being extended to address the complexities of such heterogeneous networks. Additionally, integrating services like content delivery networks (CDNs) could facilitate predictive caching, reducing bandwidth consumption for multimedia applications. Expanding SEC to support IoT-driven use cases, such as smart grids or disaster monitoring, and incorporating adaptive learning mechanisms, such as online learning or meta-learning, would enable satellites to improve their decision-making capabilities over time. Privacy-preserving techniques and resilience against adversarial attacks are crucial for secure and robust deployments. Finally, collaboration with industry partners and satellite operators to conduct real-world deployments and validations will be essential for assessing performance under practical conditions and bridging the gap between theoretical advancements and real-world applicability.

ACKNOWLEDGMENTS

This work was supported by Science and Technology Project of the State Grid Corporation of China (5400-202255158A-1-1-ZN).

REFERENCES

[1] Samuel Burer and Adam N Letchford. 2012. Non-convex mixed-integer nonlinear programming: A survey. *Surveys in Operations Research and Management Science* 17, 2 (2012), 97–106.

[2] Kaan Çelikbilek, Zainab Saleem, Ruben Morales Ferre, Jaan Praks, and Elena Simona Lohan. 2022. Survey on optimization methods for leo-satellite-based networks with applications in future autonomous transportation. *Sensors* 22, 4 (2022), 1421.

[3] Lei Cheng, Gang Feng, Yao Sun, Shuang Qin, Feng Wang, and Tony QS Quek. 2024. Energy-constrained Satellite Edge Computing for Satellite-Terrestrial Integrated Networks. *IEEE Transactions on Vehicular Technology* (2024).

[4] Weidong Fang, Chunsheng Zhu, and Wuxiong Zhang. 2023. Toward Secure and Lightweight Data Transmission for Cloud-Edge-Terminal Collaboration in Artificial Intelligence of Things. *IEEE Internet of Things Journal* (2023).

[5] Fares Fourati and Mohamed-Slim Alouini. 2021. Artificial intelligence for satellite communication: A review. *Intelligent and Converged Networks* 2, 3 (2021), 213–243.

[6] Juan A Fraire, Oana Iova, and Fabrice Valois. 2022. Space-terrestrial integrated Internet of Things: Challenges and opportunities. *IEEE Communications Magazine* (2022).

[7] Honghao Gao, Binyang Qiu, Ye Wang, Si Yu, Yueshen Xu, and Xinheng Wang. 2023. TBDB: Token Bucket-Based Dynamic Batching for Resource Scheduling Supporting Neural Network Inference in Intelligent Consumer Electronics. *IEEE Transactions on Consumer Electronics* (2023).

[8] Honghao Gao, Xuejie Wang, Wei Wei, Anwer Al-Dulaimi, and Yueshen Xu. 2023. Com-DDPG: task offloading based on multiagent reinforcement learning for information-communication-enhanced mobile edge computing in the internet of vehicles. *IEEE Transactions on Vehicular Technology* (2023).

[9] Yuanyuan Hao, Zhengyu Song, Zhong Zheng, Qian Zhang, and Zhongyu Miao. 2023. Joint Communication, Computing, and Caching Resource Allocation in LEO Satellite MEC Networks. *IEEE Access* 11 (2023), 6708–6716.

[10] Shuxin He, Tianyu Wang, and Shaowei Wang. 2020. Load-aware satellite handover strategy based on multi-agent reinforcement learning. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 1–6.

[11] Ying He, Yuhang Wang, F Richard Yu, Qiuzhen Lin, Jianqiang Li, and Victor CM Leung. 2021. Efficient resource allocation for multi-beam satellite-terrestrial vehicular networks: A multi-agent actor-critic method with attention mechanism. *IEEE Transactions on Intelligent Transportation Systems* 23, 3 (2021), 2727–2738.

[12] Xin Hu, Xianglai Liao, Zhijun Liu, Shuaijun Liu, Xin Ding, Mohamed Helaoui, Weidong Wang, and Fadhel M Ghannouchi. 2020. Multi-agent deep reinforcement learning-based flexible satellite payload for mobile terminals. *IEEE Transactions on Vehicular Technology* 69, 9 (2020), 9849–9865.

[13] Min Jia, Liang Zhang, Jian Wu, Qing Guo, Guowei Zhang, and Xuemai Gu. 2024. Deep Multi-Agent Reinforcement Learning for Task Offloading and Resource Allocation in Satellite Edge Computing. *IEEE Internet of Things Journal* (2024).

[14] Chunxiao Jiang and Xiangming Zhu. 2020. Reinforcement learning based capacity management in multi-layer satellite networks. *IEEE Transactions on Wireless Communications* 19, 7 (2020), 4685–4699.

[15] Weiwei Jiang. 2023. Software defined satellite networks: A survey. *Digital Communications and Networks* 9, 6 (2023), 1243–1264.

[16] Weiwei Jiang, Haoyu Han, Miao He, and Weixi Gu. 2024. When game theory meets satellite communication networks: A survey. *Computer Communications* 217 (2024), 208–229.

[17] Weiwei Jiang, Yafeng Zhan, Shen Xi, Defeng David Huang, and Jianhua Lu. 2021. Compressive Sensing-Based 3-D Rain Field Tomographic Reconstruction Using Simulated Satellite Signals. *IEEE Transactions on Geoscience and Remote Sensing* 60 (2021), 1–13.

[18] Weiwei Jiang, Yafeng Zhan, and Xiaolong Xiao. 2023. Multi-Domain Network Slicing in Satellite–Terrestrial Integrated Networks: A Multi-Sided Ascending-Price Auction Approach. *Aerospace* 10, 10 (2023), 830.

[19] Weiwei Jiang, Yafeng Zhan, Xiaolong Xiao, and Guanglin Sha. 2023. Network Simulators for Satellite-Terrestrial Integrated Networks: A Survey. *IEEE Access* 11 (2023), 98269–98292.

[20] Weiwei Jiang, Yafeng Zhan, Guanming Zeng, and Jianhua Lu. 2022. Probabilistic-forecasting-based admission control for network slicing in software-defined networks. *IEEE Internet of Things Journal* 9, 15 (2022), 14030–14047.

[21] Jian Jiao, Shaohua Wu, Rongxing Lu, and Qinyu Zhang. 2021. Massive access in space-based Internet of Things: Challenges, opportunities, and future directions. *IEEE Wireless Communications* 28, 5 (2021), 118–125.

[22] Ju-Hyung Lee, Jihong Park, Mehdi Bennis, and Young-Chai Ko. 2020. Integrating LEO satellite and UAV relaying via reinforcement learning for non-terrestrial networks. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 1–6.

[23] Peisong Li, Ziren Xiao, Xinheng Wang, Kaizhu Huang, Yi Huang, and Honghao Gao. 2023. EPtask: Deep reinforcement learning based energy-efficient and priority-aware task scheduling for dynamic vehicular edge computing. *IEEE Transactions on Intelligent Vehicles* (2023).

[24] Xianglai Liao, Xin Hu, Zhijun Liu, Shijun Ma, Lexi Xu, Xiuhua Li, Weidong Wang, and Fadhel M Ghannouchi. 2020. Distributed intelligence: A verification for multi-agent DRL-based multibeam satellite resource allocation. *IEEE Communications Letters* 24, 12 (2020), 2785–2789.

[25] Zhiyuan Lin, Zuyao Ni, Linling Kuang, Chunxiao Jiang, and Zhen Huang. 2022. Dynamic beam pattern and bandwidth allocation based on multi-agent deep reinforcement learning for beam hopping satellite systems. *IEEE Transactions on Vehicular Technology* 71, 4 (2022), 3917–3930.

[26] Jiahua Liu, Weiwei Jiang, Haoyu Han, Miao He, and Weixi Gu. 2023. Satellite internet of things for smart agriculture applications: A case study of computer vision. In *2023 20th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 66–71.

[27] Zhikai Liu, Navneet Garg, and Tharmalingam Ratnarajah. 2023. Multi-agent federated reinforcement learning strategy for mobile virtual reality delivery networks. *IEEE Transactions on Network Science and Engineering* (2023).

[28] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.

[29] Tarek Naous, May Itani, Mariette Awad, and Sanaa Sharafeddine. 2023. Reinforcement learning in the sky: A survey on enabling intelligence in ntn-based communications. *IEEE Access* (2023).

[30] Thuy Ngoc Nguyen, Duy Nhat Phan, and Cleotilde Gonzalez. 2023. Learning in Cooperative Multiagent Systems Using Cognitive and Machine Models. *ACM Transactions on Autonomous and Adaptive Systems* 18, 4 (2023), 1–22.

[31] Flor G Ortiz-Gomez, Daniele Tarchi, Ramón Martínez, Alessandro Vanelli-Coralli, Miguel A Salas-Natera, and Salvador Landeros-Ayala. 2021. Cooperative multi-agent deep reinforcement learning for resource management in full flexible VHTS systems. *IEEE Transactions on Cognitive Communications and Networking* 8, 1 (2021), 335–349.

[32] Zeyu Qin, Haipeng Yao, and Tianle Mai. 2020. Traffic optimization in satellites communications: A multi-agent reinforcement learning approach. In *2020 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 269–273.

[33] Jiachen Sun, Xu Chen, Zhen Li, Jiawei Wang, and Yuxi Chen. 2024. Joint Optimization of Multiple Resources for Distributed Service Deployment in Satellite Edge Computing Networks. *IEEE Internet of Things Journal* (2024).

[34] Jin Tang, Jian Li, Lan Zhang, Kaiping Xue, Qibin Sun, and Jun Lu. 2022. Content-Aware Routing based on Cached Content Prediction in Satellite Networks. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 6541–6546.

[35] Bo Wang, Jiecheng Xie, and Dongyan Huang. 2024. Computation offloading strategies for LEO satellite edge computing systems based on different multiple access methods. *IEEE Access* (2024).

[36] Hongbign Wang, Xin Chen, Qin Wu, Qi Yu, Xingguo Hu, Zibin Zheng, and Athman Bouguettaya. 2017. Integrating reinforcement learning with multi-agent techniques for adaptive service composition. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 12, 2 (2017), 1–42.

[37] Yanting Wang, Min Sheng, Xijun Wang, Liang Wang, and Jiandong Li. 2016. Mobile-edge computing: Partial computation offloading using dynamic voltage scaling. *IEEE Transactions on Communications* 64, 10 (2016), 4268–4282.

[38] Ying Wang, Yichun Xu, Yuan Zhang, and Ping Zhang. 2017. Hybrid satellite-aerial-terrestrial networks in emergency scenarios: A survey. *China Communications* 14, 7 (2017), 1–13.

[39] Zhibo Xing, Mingxia Huang, and Dan Peng. 2023. Overview of machine learning-based traffic flow prediction. *Digital Transportation and Safety* 2, 3 (2023), 164–175.

[40] Rui Xu, Xiaoqiang Di, Jing Chen, Haowei Wang, Hao Luo, Hui Qi, Xiongwen He, Wenping Lei, and Shiwei Zhang. 2023. A hybrid caching strategy for information-centric satellite networks based on node classification and popular content awareness. *Computer Communications* 197 (2023), 186–198.

[41] Zhaohui Yang, Cunhua Pan, Kezhi Wang, and Mohammad Shikh-Bahaei. 2019. Energy efficient resource allocation in UAV-enabled mobile edge computing networks. *IEEE Transactions on Wireless Communications* 18, 9 (2019), 4576–4589.

[42] Hongwei Zeng, Zhongzhi Zhu, Ye Wang, Zhengzhe Xiang, and Honghao Gao. 2024. Periodic Collaboration and Real-Time Dispatch Using an Actor–Critic Framework for UAV Movement in Mobile Edge Computing. *IEEE Internet of Things Journal* (2024).

[43] Changzhen Zhang and Jun Yang. 2024. An Energy-Efficient Collaborative Offloading Scheme With Heterogeneous Tasks for Satellite Edge Computing. *IEEE Transactions on Network Science and Engineering* (2024).

Received 22 February 2024; revised 17 November 2024; accepted 21 January 2025