Hierarchical DRL-empowered Network Slicing in Space-Air-Ground Networks

Hierarchical DRL-empowered Network Slicing in Space-Air-Ground Networks

Abegaz Mohammed Seid*, Hayla Nahom Abishu †, Aiman Erbad*, and Carla Fabiana Chiasserini‡
*Division of Information and Computing Technology, College of Science and Engineering
Hamad Bin Khalifa University, Doha, Qatar
†School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu
‡ Politecnico di Torino, Italy
Email: *{mabegaz, AErbad}@hbku.edu.qa, †nahomh185@gmail.com, ‡carla.chiasserini@polito.it

***Abstract*—The space-air-ground integrated network (SAGIN) is an emerging architecture that has the potential to provide seamless, high data rates, and reliable transmission with vastly increased coverage for intelligent edge devices (iEDs). However, the SAGIN infrastructure is quite complex consisting of multiple network segments; it is thus critical to efficiently manage the network segments' resources to ensure QoS satisfaction (e.g., delay and rate) for the various services provided to the iEDs. In this regard, network slicing (NS) and overall network softwarization technologies can play an essential role in addressing iEDs QoS and utility needs. In this work, we propose an optimal intelligent end-to-end resource allocation with network slicing in multi-tier SAGIN to maximize the network performance. We model the network depending on its service requirements. As the above optimization problem turns out to be NP-hard, we transform it into a stochastic game model and efficiently solve it through hierarchical multi-agent deep reinforcement learning (HMADRL). In particular, we decompose it into two parts, i.e., optimizing the mapping combined with slice adjustment and the resource allocation with association problem. Both problems are then solved using multi-agent DRL. The simulation results demonstrate that our proposed HMADRL algorithm outperforms the baseline algorithms in terms of maximizing the utility and QoS satisfaction of iEDs.**

***Index Terms*—Network slicing, SAGIN, 6G, DRL, Resource allocation, QoS**

I. Introduction

The space-air-ground integrated network (SAGIN) is one of the key enablers of the upcoming 6G networks and is expected to support diversified vertical applications, ranging from monitoring and reconnaissance to in-space backhauling. SAGIN is therefore required to meet the stringent and diverse quality of service (QoS) that such applications demand. As highlighted in [1], this is a critical, open issue, since current network slicing (NS) techniques, developed for 5G networks, cannot fully satisfy the stringent and diversified QoS requirements of the 6G emerging use cases.

Interestingly, some recent work has discussed the role of service virtualization and software-defined networking (SDN) in SAGIN, tacking different aspects. The authors of [2] proposed an automatic NS framework for a space-ground integration to place computations and perform resource allocation with minimal service level agreement (SLA) violations. Lyu *et al.* [3], proposed a SAGIN framework to provide virtualized and micro-services to terrestrial users to achieve efficient, flexible and scalable online service provisioning. In [4], the authors emphasized the advantages of SDN in space-air-integrated vehicular networks and identified several difficulties, such as dynamic networking, QoS provisioning, interoperation, and network management. In SAGIN, they examined the importance of artificial intelligence (AI) for effective NS, mobility management, and other optimization issues. The study in [5] presents, instead, a dynamic resource slice framework in SAGIN for the provision of isolated vehicular services, with the aim to maximize the revenue. Similarly, [6] proposes a dynamic SDN/virtual network functions (VNF) mapping and scheduling framework for SAGIN, to facilitate the delivery of E2E IoV services and maximize the profit of the service provider. Finally, [7] introduces deep reinforcement learning (DRL)-based multi-domain resource allocation in a SAGIN infrastructure, using virtualization to maximize revenue.

It is worth noting that E2E NS combines cloud native 5G features to create agile and flexible demand-driven systems that respond quickly and efficiently to customer needs [8]. Nonetheless, the traditional cloud computing-assisted NS may pose significant challenges in satisfying the ultra-responsiveness and high energy efficiency requirements of intelligent edge device (iED) applications in SAGIN, as many of such applications exhibit more and more stringent requirements, in terms of latency, data security, and reliability. To overcome this issue, it is critical to leverage hierarchical multi-tier computing (MTC) [9], one of the essential paradigms in 6G networks, as it enables cooperative computing across the three segments of the SAGIN using MEC servers deployed in space, air, and ground networks.

Most of the existing work on virtualized resource allocation in the SAGIN architecture has focused on a single network domain NS, which can besolved by, e.g., heuristic algorithms [5], [6] and traditional DRL approaches [7]. However, the SAGIN infrastructure requires multidomain E2E NS to map a high-level description of services into a tangible E2E slice. In addition, numerous factors need to be considered in the SAGIN infrastructure, including route scheduling, traffic distribution, power control, spectrum allocation, and load balancing. Thus, efficient resource allocation and scheduling of multidomain physical infrastructure are essential. In such hierarchical multidomain networks, resource allocation could be handled by intelligent and flexible orchestration systems such as virtual network embedding and SDN [4]. The former technique can indeed effectively solve the SAGIN resource allocation and scheduling problem. However, a multidomain E2E network slicing algorithm necessitates hierarchical intelligent dynamic resource management based on AI methods to control the admission of various resource requests from various iED and virtual network operators, as well as resource adjustment within admitted slices for each tenant. Moreover, utilizing AI for resource management in the 6G network allows the virtual network operators to make intelligent slice admission decision.

In this work, we consider an E2E NS for multidimensional resource allocation in SAGIN, which depends on slice-level feedback and real-time slice adaptation to maximize the utility of the infrastructure as well as ensure the QoS required by iEDs applications. We propose a hierarchical multi-agent DRL (HMADRL)-enabled resource allocation to ensure the deployment of isolated services. In each time slot, the framework allows the agents to make intelligent decisions, such as managing the number of requests from different services in a separate layer to provide admission and associate the requests with satellite, HAPS/UAV, and ground nodes to allocate resources. To the best of our knowledge, this is the first study to apply a hierarchical DRL framework to optimize E2E NS and resource allocation across multiple domains in the SAGIN infrastructure. The framework in this work consists of two levels: high-level policy is used to determine the mapping and network slicing strategies, while low-level policy optimizes the resource allocation. Therefore, the proposed framework provides a flexible and adaptive solution to the multidimensional resource allocation problem in the SAGIN infrastructure. Our main contributions are as follows:

- We propose a distributed MADRL-enabled E2E NS in MTC-enabled SAGIN infrastructure that focuses on use cases of eMBB, mMTC, and uRLLC concerning priorities such as radio resource allocation, computing resource allocation, and latency management.
- We formulate the optimization problem that maximizes the utility of all three E2E network slices by jointly optimizing the allocation of communication and computing resources while satisfying the QoS requirements of iEDs.
- Since the formulated optimization problem is non-convex and difficult to solve directly via conventional optimization techniques, we transform it into a stochastic game model and solve it using the HMADRL algorithm.
- Finally, exhaustive simulations are conducted to validate the performance of the proposed framework.

II. System Model

The system we address, shown in Fig.1, consists of the three segments of the physical SAGIN (i.e., space, air, and ground), and it includes both the terrestrial and non-terrestrial mobile network operators (T-MNO and NT-MNO, resp.), which provide the communication infrastructure and the related computing, bandwidth, and storage resources to iED users. The space segment of the SAGIN consists of low-earth orbit (LEO) satellites that can cover large geographical areas and connect with the lower-segment networks via wireless links. The aerial segment includes HAPS and UAVs and communicates with space and terrestrial networks through wireless backhaul links. The ground segment consists of terrestrial base stations (TBS) interconnected with each other and providing resources to iEDs in normal situations. However, whenever the TBS loses connection with the iEDs and upper layer network segments, the iEDs can be associated with an aerial segment or a space segment, depending on the quality of the connectivity and the availability of resources. The physical resources of the SAGIN infrastructure, managed by the infrastructure owners, are virtualized and segmented into slices, which are then leased to T-MNOs and NT-MNOs; the latter can then assign these slices to tenants with certain SLAs.

Due to the nature of the SAGIN architecture, the SDN controller is physically distributed across the different layers to meet the E2E NS and resource allocation requirements.

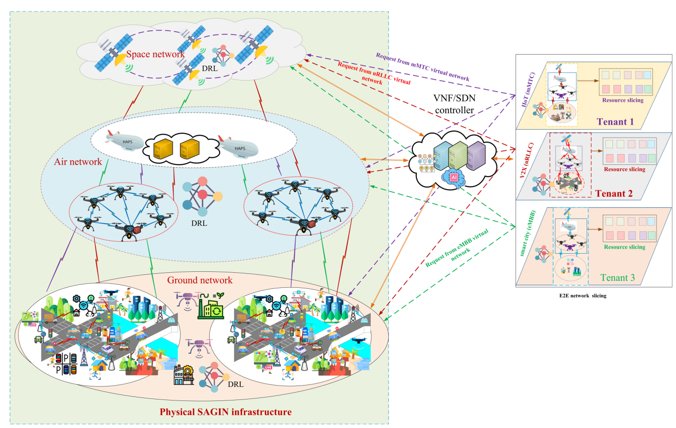


Fig. 1: System Model.

In this work, we model the physical SAGIN infrastructure by a graph $G = \{\mathcal{N}, \mathcal{L}, \mathcal{A}\}$, where $\mathcal{N} = \mathcal{J} \cup \mathcal{H} \cup \mathcal{K}$ is the set of physical nodes (space $\mathcal{J}$, air $\mathcal{H}$, and terrestrial $\mathcal{K}$ nodes), $\mathcal{L} = \{j, h, k, (j,h), (j,k), (h,k)\}$ is the set of communication links (space, air, terrestrial, space-air (S2A), space-terrestrial (S2T) and air-terrestrial (A2T) communication links) and $\mathcal{A} = \{F_j, F_h, F_k, B_j, B_h, B_k, T_i\}$ is the set of network attributes/resources including computing resources F_j, F_h, and F_k and bandwidth resources B_j, B_h, and B_k of space, air, and ground links, respectively. For efficient allocation of communication and computing resources in the hierarchical

SAGIN architecture, we implement the E2E NS technique with dynamic slice creation and related mapping onto the physical infrastructure. To this end, we combine the SDN and NFV technologies with machine learning algorithms, yielding efficient E2E NS and resource allocation in the SAGIN.

A. Virtual Network Model

We consider that distributed VNF and SDN are implemented at each physical layer so as to dynamically control NSs, slice requests, and resource allocations. Let $\xi = \{\xi_e, \xi_u, \xi_m\}$ denote the set of slices in the MTC-enabled SAGIN architecture, where ξ_e, ξ_m, and ξ_u, represent eMBB (smart city), mMTC (IIoT), and uRLLC (V2X) slices, respectively. Each slice is deployed over the space, air, and terrestrial networks, and each slice exhibits different E2E latency requirements. Slice requests, $\mathcal{Q}_{n^v}$, are modeled by graph $G^v = \{\mathcal{N}^v, \mathcal{L}^v, \mathcal{A}^v\}$, where $\mathcal{N}^v$ is the set of virtual nodes, $\mathcal{L}^v$ is the set of virtual communication links, and $\mathcal{A}^v$ is an attribute of the set of slice requests. Here, $\mathcal{A}^v = \{F_j^v, F_h^v, F_k^v, B_j^v, B_h^v, B_k^v, B_{jh}^v, B_{jk}^v, B_{hk}^v, T_i^v\}$, where F_j^v, F_h^v, and F_k^v denote the set of slice requests in the space, air, and ground segment, respectively, and $B_j^v, B_h^v, B_k^v, B_{jh}^v, B_{jk}^v$, and B_{hk}^v denote the bandwidth required on the various types of network links. Also, T_i^v denotes the task of iED i defined as $T_i^v = \{\phi_i, \varsigma_i, \tau^{max}, \vartheta_i\}$, where $\phi_i, \varsigma_i, \tau^{max}$, and ϑ_i are the input size of the intensive computing task, the CPU frequency required to complete the input, the maximum tolerable latency of the input, and the type of admission of service for uRLLC, eMBB, or mMTc slices, respectively. As for the core network controller, there is a binary variable $\Phi_{n^v}(t) \in \{0, 1\}$ that identifies the mapping of a physical node with a virtual node in a particular slice: $\Phi_{n^v}(t) = 1$ if the requested slice q_{n^v} uses physical node n, otherwise $\Phi_{n^v} = 0$.

Next, we consider that, given the MTC-enabled SAGIN architecture, resource providers and service providers have to allocate resources to a set of ultra-dense heterogeneous iEDs, $\mathcal{I} = \{1, 2, \ldots, I\}$ so as to meet the diverse QoS requirements for the different slices. The set of iEDs subscribed to tenant ξ, associated with network node n, can be defined as $\alpha_{in}(t) \in \{-1, 0, 1\}$, where $\alpha_{in}(t) = -1$ denotes that iED i associated with the ground node; $\alpha_{in}(t) = 0$ if iED i associated with air node, and when $\alpha_{in}(t) = 1$, iED i associated with space node. Each iED i is assumed to be associated with a single network operator, i.e., the one with more resources or higher connectivity quality in a given time slot t. The allocation of resources (bandwidth and computing) to slice ξ in physical node $n \in \mathcal{N} = \{1, 2, \ldots, N\}$ and the associated virtualized node $n^v \in \mathcal{N}^v = \{1, 2, \ldots, N^v\}$, is given by: $B_{n,n^v}^{\xi} = \Phi_{n^v}(t)\alpha_{in}(t)\sum_{n=1}^{N}\sum_{n^v=1}^{N^v} b_{n,n^v}^{\xi}$ and $F_{n,n^v}^{\xi} = \Phi_{n^v}(t)\alpha_{in}(t)\sum_{n=1}^{N}\sum_{n^v=1}^{N^v} f_{n,n^v}^{\xi}$, where b_{n,n^v}^{ξ} and f_{n,n^v}^{ξ} are the bandwidth and computing resources allocated to iED i in slice ξ in (n, n^v).

The achievable spectrum efficiency of iED i through the physical resource block of node n is expressed as $\Gamma_{in,n^v}^{\xi}(t) = log_2\big(1+\frac{p_{in,n^v}^{\xi}|h_{in,n^v}^{\xi}|^2}{\delta^2+I_{n,n^v}^{\xi}}\big)$, where p_{in,n^v}^{ξ} and h_{in,n^v}^{ξ} represent the transmission power and channel gain of the iED i associated with the linked physical node n and virtual node n^v. The data rate of iED i associated with node (n, n^v) in the ξ slice is $r_{in,n^v}(t) = \Phi_{n^v}(t)\alpha_{in}(t)\upsilon_{in,n^{\xi}}B_{n,n^v}\Gamma_{in,n^v}$, where $\upsilon_{in,n^v} \in [0, 1]$ is the bandwidth fraction allocated from node n^v to iED i in the slice ξ. Furthermore, the computing capability of iED i is indicated as $f_{in,n^v}(t) = \rho_{in,n^{\xi}}F_{n,n^v}$, where $\rho_{in,n^v} \in [0, 1]$ represents the fraction of the computing capability in the linked node (n, n^v) allocated to the iED i in the slice ξ.

Let us then define the binary admission decision variable $\psi_{in,n^v}^{\xi}(t) = \{0, 1\}$, where $\psi_{in,n^v}^{\xi}(t) = 1$ indicates that iED i's service request gets admitted to use slice ξ at network link node(n, n^v) at time slot t. For any admitted iED i, the minimum allocation of resources to the iED is denoted with $i^{\xi} = [b_{in,n^v}^{\xi,min}, f_{in,n^v}^{\xi,min}]$, where $b_{in,n^v}^{\xi,min}$ and $f_{in,n^v}^{\xi,min}$ are bandwidth and computing resources (resp.), as per the related SLA. If the SLAs cannot be met, iEDs are not allowed to use the slice. The maximum number of iEDs that can be allowed in slice ξ at node (n, n^v) is given by: $|\mathcal{I}_{n,n^v}^{\xi}|_{min} = min\Big\{\frac{b_{n,n^v}^{\xi}}{b_{in,n^v}^{\xi,min}}, \frac{f_{n,n^v}^{\xi}}{f_{in,n^v}^{\xi,min}}\Big\}$. Then, the system can admit a specific request or all requests, depending on the availability of resources ($0 \le |\mathcal{I}_{n,n^v}^{\xi}|_{min} \le |\mathcal{I}_{n,n^v}^{\xi}|_{max}$). When a slice request is admitted and associated with a single virtual node n^v in time slot t, $\sum_{n^v \in \mathcal{N}^v} \Phi_{n^v}\psi_{in,n^v}^{\xi}(t)\alpha_{in}(t) = 1$. The physical resources occupied by the associated slice constraint in terms of computing and bandwidth resource are expressed as:

$$\sum_{q=1}^{|Q_{n,n^v}^{\xi}|} \sum_{n \in \mathcal{N}} \sum_{T_i^{\xi} \in \mathcal{A}^{\xi}} \psi_{in,n^v}^{\xi}(t)\alpha_{in}(t)(t)f_{in,n^v}^{\xi}(t) \le F_n. \quad (1)$$

$$\sum_{q=1}^{|Q_{n,n^v}^{\xi}|} \sum_{n \in \mathcal{N}} \sum_{T_i^{\xi} \in \mathcal{A}^{\xi}} \psi_{in,n^v}^{\xi}(t)\alpha_{in}(t)(t)b_{in,n^v}^{\xi}(t) \le B_n. \quad (2)$$

The general demand for Z_{N,N^v}^{ξ} of slice ξ is given by: $Z_{N,N^v}^{\xi} = \sum_{q_{\xi}=1}^{|Q_{\xi}|} \alpha_{in}(t)\mathcal{A}_{n,n^v}^{\xi}$. The contribution of a specific slice ξ to the system traffic is instead: $\tilde{Z}_{N,N^V}^{\xi} = \frac{|Q_{n,n^v}^{\xi}|}{\sum_{\xi=1}^{\xi}|Q_{n,n^v}^{\xi}|}$.

When iED i is associated with the virtual node n^v in slice ξ and offloads the intensive task T_i^{ξ} to node n through the allocated channel and radio resource, the task completion latency is given by: $T_{in,n^v}^{\xi}(t) = \max\Big\{\frac{\phi_i\varsigma_i}{\rho_{in,n^v}^{\xi}F_{n,n^v}^{\xi}}, \max\Big\{\frac{\phi_i}{\nu_{in,n^V}^{\xi}B_{n,n^v}^{\xi}}, \frac{\phi_i}{r_{in}^{\xi}(t)}\Big\}+\tilde{T}\Big\}+\tilde{T}$, where $\tilde{T}$ denotes the time interval. Then the QoS satisfaction of iED i in a specific slice ξ can be defined as [10]: $\Phi_{in,n^v}^{\xi}(t) = \frac{1}{1+e^{-\iota(\rho_{in,n^v}^{\xi}f_{in,n^v}^{\xi}(t)|\nu_{in,n^v}^{\xi}b_{in,n^v}^{\xi}-q_{in^v}^{\xi}(t))}}$, where ι and $q_{in^v}(t)$ represent the constant value and the slice demand in a specific slice ξ when iED i is associated with a particular node n in time slot t.

B. Utility model and problem formulation

We now define the utility of the nodes that provide virtual resources in different slices. It is defined as the difference between the revenue and the cost of task computing in a given time slot and at different virtualized nodes, while considering the iEDs satisfaction rate. The revenue of a specific task depends on the minimization of the latency when executed on a given slice:

$$U^{\xi}_{in,n^v}(t) = \Big(\omega^{\xi,b}_{in,n^v}(\upsilon^{\xi}_{in,n^v} B^{\xi}_{n,n^v}) + \omega^{\xi,f}_{in,n^v}(\rho^{\xi}_{in} F^{\xi}_{n})\Big) - \varrho^{\xi}_{in,n^v} C^{\xi}_{in,n^v}, \tag{3}$$

where $\omega^{\xi,b}_{in,n^v}$ and $\omega^{\xi,f}_{in,n^v}$ denote the unit price of the bandwidth and computing resources allocated to node n, n^v for slice ξ. Also, ϱ^{ξ}_{in,n^v} and C^{ξ}_{in,n^v} denote the costs (in terms of computing energy and other related costs, resp.) of C^{ξ}_{in,n^v} of node n, n^v in slice ξ. Therefore, the joint optimization problem is given as:

$$P_1 \quad \max_{\alpha,\psi,\upsilon,\rho} \sum_{n^{v\,\xi} \in \mathcal{N}^{v\,\xi}} \sum_{\xi=1}^{\xi} U^{\xi}_{in,n^v}(t) \tag{4a}$$

$$\text{s.t.} \quad \textbf{C1}: \sum_{n,n^{v\,\xi} \in \mathcal{N}^{v\,\xi}} \alpha_{in}(t) \leq 1, i \in \mathcal{I}^{\xi}_{n,n^v} \tag{4b}$$

$$\textbf{C2}: \sum_{n,n^{v\,\xi} \in \mathcal{N}^{v\,\xi}} \psi^{\xi}_{in,n^v}(t) \leq 1, i \in \mathcal{I}^{\xi}_{n,n^v} \tag{4c}$$

$$\textbf{C3}: \sum_{n,n^{v\,\xi} \in \mathcal{N}^{v\,\xi}} \alpha_{in}(t)(t)\psi^{\xi}_{in,n^v}(t)\rho^{\xi}_{in,n^v} \leq 1 \tag{4d}$$

$$\textbf{C4}: \sum_{n,n^{v\,\xi} \in \mathcal{N}^{v\,\xi}} \alpha_{in}(t)(t)\psi^{\xi}_{in,n^v}(t)\nu^{\xi}_{in,n^v} \leq 1 \tag{4e}$$

$$\textbf{C5}: (1), (2) \tag{4f}$$

$$\textbf{C6}: \alpha_{in}(t), \psi^{\xi}_{in,n^v}(t) \in \{0, 1\} \tag{4g}$$

$$\textbf{C7}: \rho^{\xi}_{in,n^v}(t), \nu^{\xi}_{in,n^v}(t) \in [0, 1] \tag{4h}$$

$$\textbf{C8}: \Phi^{\xi}_{in,n^v}(t) > 0.5 \tag{4i}$$

Constraints $C1$ and $C2$ indicate the association between an iED and a virtual node in a particular slice. The admission of an iED in a virtual system to access resources in a given slice depends on the type of service. $C3$ and $C4$ specify that each virtual node's fractional computing and bandwidth capacities should be less than or equal to one, respectively. $C5$ imposes that the bandwidth allocated and computing resources should not exceed the maximum available resources at the physical nodes. $C6$ indicates the binary decision variables of association and admission. $C7$ restrictslimits the fractional variables of bandwidth and computing resources. $C8$ states that the QoS satisfaction of the iED should be greater than 0.5.

III. HMADRL-BASED SOLUTION

The dynamic E2E NS and efficient resource allocation in MTC-enabled SAGIN infrastructure supports intelligent decision-making in a hierarchical and dynamic environment to maximize long-term utility. The optimization problem given in Eq. (4) is NP-hard due to cross-domain resource allocation with varying characteristics and constraints, heterogeneity in resource demands, and a large search space in the dynamic SAGIN infrastructure to obtain optimal resource allocation and mapping. Intuitively, the problem complexity comes from the objective function that aims to maximize utility across different layers while ensuring the QoS requirements of iEDs.

We therefore develop the HMADRL framework, which performs problem decomposition and solves the following two multi-objective optimization problems (i) it optimizes the mapping between physical and virtual resources, as well as slice adjustment, and (ii) it allocates resources so as to maximize the system's utility while ensuring the QoS of different iEDs. To do so, it leverages a model-free DDPG approach [11] for the first problem, and the MADRL approach [12] for the second one. The above approaches are implemented thanks to the SDN controller that creates maps of physical and virtual nodes, as well as multiple agents, including space, air, and ground nodes, that allocate resources to iEDs based on service type, admission capability, and demand.

Algorithm 1 : HMADRL for E2E NS resource allocation

Initialize: Experience replay buffer at controller $\mathcal{D}^{controller}$, and $\mathcal{D}^{node}$
Initialize: The parameters of actor and critic network with random weights θ;
for episode = 1 to 3000 **do**
  Reset the environment
  Update mapping and slice adjustment action a_t by DDPG Algorithm
  All agents observe the initial state $S = \{s_1, s_2, ..., s_N\}$
  for t=1 to 200 **do**
    Each agent n selects a random action a_n based on the probability ε, else select action $a_n = \pi_{\theta_l}(s_n)$;
    All agents execute action $a(t)$ and observe rewards $r(t)$, and the new state $s_n(t+1) \sim s'_n$
    Store the tuples $\{s_n(t), a_n(t), r_n(t), s'_n(t+1)\}$
    $s_n \leftarrow s'_n$;
    for agent $n = 1$ to N **do**
      Randomly select mini-batch of H samples tuples
      Update the critic-network by minimizing the loss
      Update actor-network using sample policy gradient
    end for
    Update target network parameters for each agent n
  end for
end for

A. DRL-enabled mapping

A) State Space: The state represents the available resource blocks at the physical nodes. A state, $s \in S$, is represented by $s_t = \{F, F_n, B_{n,n^v}\}$, where F, F_n and B_{n,n^v} are the available computing resource blocks on the core network, available computing resource blocks of nodes in space, air, and ground segments, and the communication resource block in the set of links. $F = 0.8$, $F_n = \{0.6, 0.55, 0.65\}$ represent 80%, 60%, 55% and 65% of the overall available computing capacity of the main computing server, the ground (base station), the air (HAPS/UAV) and the space computing servers (resp.), while $B_{n,n^v} = 0.8$ denotes the available computational resources in the links.

B) Action: The DRL agent creates aslice based on the type of services to support, the service demand, and the associated latency requirements. An action $a_t \in A$ is specified by $a_t = \{\omega^u \xi_u, \omega^e \xi_e, \omega^m \xi_m\}$, where ω^u, ω^e and ω^m (s.t. $\omega^u + \omega^e + \omega^m = 1$) define the weight of URRL, eMBB, and mMTC slices.

C) Reward function: As the goal account is to maximize the number of iED served by the system by optimally mapping physical and virtual resources while meeting the different slices requirements, the agent reward is: $R = \sum_{n=1}^{N} \sum_{n^v=1}^{N^v} r_t$, with , $r_t = \begin{cases} 1 & \text{if } \Phi_{n^v}(t) = 1 \\ 0 & \text{otherwise} \end{cases}$

B. MADRL-enabled E2E resource allocation

A) State space: The state $s_n(t) \in S$ of the agents in each slice is defined as $s_n(t) = \{T_i^v, \tilde{b}_i^{v,\xi}, \tilde{f}_i^{v,\xi}, \omega_{in,n^v}^{\xi,b}, \omega_{in,n^v}^{\xi,f}, \varrho_{in,n^v}^{\xi}, \check{f}_{n,n^v}^{\xi}, \check{b}_{n,n^v}^{\xi}\}$, where $T_i^v, \tilde{b}_i^{v,\xi}, \tilde{f}_i^{v,\xi}, \omega_{in,n^v}^{\xi,b}, \omega_{in,n^v}^{\xi,f}, \check{f}_{n,n^v}^{\xi}$, and $\check{b}_{n,n^v}^{\xi}$ indicate the task profile, communication resource demands, computing resource demand, price of fractional communication resources, price of fractional computing resources, the available computing resources, and the available communication resources, respectively. ϱ_{in,n^v}^{ξ} accounts for the costs incurred during resource allocation.

B) Action space: At each phase, a decision should be made regarding whether to admit incoming slice requests, the association between iED and nodes/links to compute-intensive tasks and relay tasks, and the allocation of resources. Hence, $a_n(t) \in A = \{a_1(t), a_2(t), \ldots, a_N(t)\}$, which can be defined as: $a_n(t) = \{\alpha_{in}(t), \psi_{in,n^v}^{\xi}(t), \nu_{in,n^v}^{x} i(t), \rho_{in,n^v}(t)\}$. The notations in the action set $\alpha_{in}(t), \psi_{in,n^v}^{\xi}(t), \nu_{in,n^v}^{x} i(t)$, and $\rho_{in,n^v}(t)$ refers to the action of association/offloading decision, iED admission, fraction of bandwidth allocation, and fraction of computing resource allocation, respectively.

C) Reward function: It represents the revenue/profit coming from all admitted slice requests while accounting for the cost of the resources allocated to satisfy the iEDs's QoS requirements in a given time slot, i.e., $R = \sum_{n=1}^{N} r_n(t)$,
$r_n(t) = \sum_{i=1}^{I} U_{in,n^v}^{\xi}(t)$.
.

IV. Numerical results

To evaluate the performance of our proposed framework, we consider 30 physical nodes with $[200, 400]$ physical links, 6 physical space nodes, 15 physical air nodes, and 9 ground nodes. The computational resources of the space, air and terrestrial nodes defined as $[50, 300]$ GHz, $[20, 200]$ GHz and $[50, 250]$ GHz, respectively, and communication resources for all segments' nodes defined as $[50, 100]$ Mbps. The delay for all links is $[1, 101]$ ms. The number of slice requests for the three services is $[50, 100]$, and there are approximately 20 virtual nodes in the system with computing and communication requirements of $[5, 10]$ GHz and $[5, 30]$ Mbps for virtual nodes. The unit price for fraction of resources is [0,1]. The DRL and MADRL configuration is as in [11], [12]. We compare the proposed solution against three baselines, namely, multi-agent DDPG (MADDPG), DDPG, and a greedy approach.

The convergence of the HMADRL algorithm and the benchmark algorithms is shown in Fig. 2. We compared the system

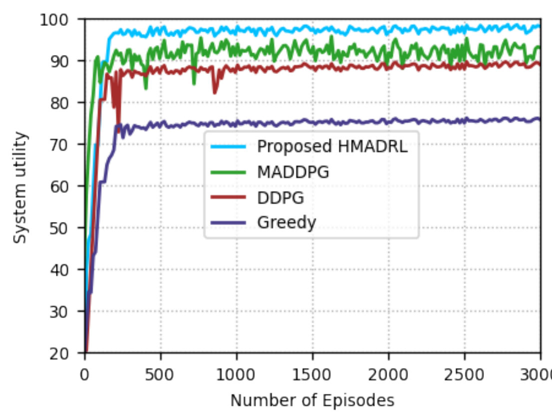
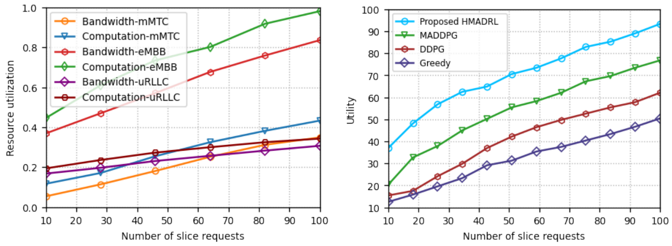


Fig. 2: System utility.

utility convergence with the increasing number of episodes. As seen in Fig. 2, the system utility converges in all algorithms at about 400 episodes. The proposed HMADRL algorithm, on the other hand, converges faster than the other algorithms. In particular, the proposed HMADRL algorithm outperforms the benchmark algorithms in system utility maximization, which is 20.45%, , 22.68%, and 56.87% better than the MADDPG, DDPG, and greedy algorithms, respectively.

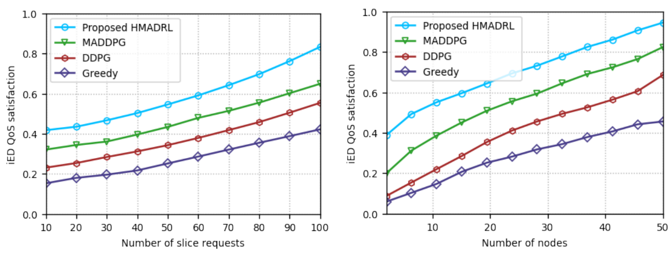
Fig. 3 illustrates the effects of slice requests of diverse service aware slices in utility and resource utilization. As the intuition suggests, Fig. 3(a) shows that the resource utilization of the proposed framework gradually increases as the number of slice requests grows. Also, eMBB services consume higher computational and bandwidth resources than uRLLC and mMTC services, implying that eMBB implies more traffic and iEDs to support than the others. Fig. 3(b) presents the system utility achieved by the proposed HMADRL compared to the baselines, for three slice types (eMBB, uRLLC, and mMTC). The system's utility increases as the slice request



(a) Resource utilization. (b) Utility.

Fig. 3: Resource utilization and utility analysis.

increases in all algorithms, as the amount of allocated resources and the price per resource fraction will increase as well. From the figure, we can observe that the utility increases more rapidly in the proposed HMADRL algorithm than in the baselines, since the HMADRL algorithm breaks down complex jobs into subtasks, thus turning out to be more efficient.



(a) (b)

Fig. 4: Analysis of iEDs' QoS Satisfactions.

Fig. 4 depicts how the number of of slice requests and the resource providers affect the QoS satisfaction of iEDs. Fig. 4(a) highlights that the QoS satisfaction of iEDs increases as the number of slice requests for all services and layers increases. Indeed, as the number of requests grows, dispensing more resources may increase latency and lower service quality. However, SAGIN's virtualized infrastructure and resources are scalable and can handle the increased demand. In particular, the proposed HMADRL algorithm can increase the QoS of iEDs satisfaction by 22.02%, 33.49%, and 49.34%, when compared to MADDPG, DDPG, and greedy, respectively. Fig. 4(b) demonstrates that the number of nodes can affect the QoS of iEDs' satisfaction in a virtualized SAGIN infrastructure. The IEDs' QoS satisfaction increases as the virtualized network nodes increase gradually in all algorithms. Clearly, adding virtual nodes to maintain user QoS helps to handle the increased load. The proposed HMADRL algorithm achieves better QoS satisfaction than MADDPG, DDPG, and greedy, by 12.8%, 27.4%, 51.75%, respectively. Also, the MADDPG algorithm outperforms the DDPG and greedy approaches in meeting theQoS requirements.

V. Conclusion

We proposed an HMADRL framework for E2E slice resource allocation in a virtualized SAGIN infrastructure. With the aim to improve iEDs' QoS satisfaction while maximizing the system's utility, we first formulated an optimization problem and then, in light of its complexity, we transformed it into a stochastic game model. To solve the latter, we applied an HMADRL algorithm that leverages a DDPG approach to solve the mapping between physical and virtual resources and slice adjustment, and a multi-agent DRL (MADDPG) to dynamically allocate resources to the different tenants. Simulation results demonstrate that the proposed HMADRL outperforms baseline solutions thanks to its ability to adapt to the system and iEDs demand changes. Our work can pave the way for future research and development in this area and contribute to ongoing efforts to enhance the performance and efficiency of SAGIN systems.

References

[1] W. Guan, H. Zhang, and V. C. M. Leung, "Customized slicing for 6g: Enforcing artificial intelligence on resource management," *IEEE Network*, vol. 35, no. 5, pp. 264–271, 2021.

[2] A. Kak and I. F. Akyildiz, "Towards automatic network slicing for the internet of space things," *IEEE Transactions on Network and Service Management*, vol. 19, no. 1, pp. 392–412, 2022.

[3] F. Lyu, F. Wu, Y. Zhang, J. Xin, and X. Zhu, "Virtualized and micro services provisioning in space-air-ground integrated networks," *IEEE Wireless Communications*, vol. 27, no. 6, pp. 68–74, 2020.

[4] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 101–109, 2017.

[5] F. Lyu, P. Yang, H. Wu, C. Zhou, J. Ren, Y. Zhang, and X. Shen, "Service-oriented dynamic resource slicing and optimization for space-air-ground integrated vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7469–7483, 2022.

[6] J. Li, W. Shi, H. Wu, S. Zhang, and X. Shen, "Cost-aware dynamic sfc mapping and scheduling in sdn/nfv-enabled space–air–ground-integrated networks for internet of vehicles," *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 5824–5838, 2022.

[7] P. Zhang, C. Wang, N. Kumar, and L. Liu, "Space-air-ground integrated multi-domain network resource orchestration based on virtual network architecture: A drl method," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2798–2808, 2022.

[8] H. Bai, Y. Zhang, Z. Zhang, and S. Yuan, "Latency equalization policy of end-to-end network slicing based on reinforcement learning," *IEEE Transactions on Network and Service Management*, vol. 20, no. 1, pp. 88–103, 2023.

[9] K. Wang, J. Jin, Y. Yang, T. Zhang, A. Nallanathan, C. Tellambura, and B. Jabbari, "Task offloading with multi-tier computing resources in next generation wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 2, pp. 306–319, 2023.

[10] M. Sheng, C. Xu, X. Wang, Y. Zhang, W. Han, and J. Li, "Utility-based resource allocation for multi-channel decentralized networks," *IEEE Transactions on Communications*, vol. 62, no. 10, pp. 3610–3620, 2014.

[11] A. M. Seid, G. O. Boateng, S. Anokye, T. Kwantwi, G. Sun, and G. Liu, "Collaborative computation offloading and resource allocation in multi-uav-assisted iot networks: A deep reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 15, pp. 12 203–12 218, 2021.

[12] A. M. Seid, G. O. Boateng, B. Mareri, G. Sun, and W. Jiang, "Multi-agent drl for task offloading and resource allocation in multi-uav enabled iot edge network," *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 4531–4547, 2021.