8-21-2022

FDRL Approach for Association and Resource Allocation in Multi-UAV Air-To-Ground IoMT Network

Abegaz Mohammed
Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

Aiman Erbad
Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

Hayla Nahom
School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China

Abdullatif Albaseer
Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

Mohammed Abdallah
Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

See next page for additional authors

FDRL Approach for Association and Resource Allocation in Multi-UAV Air-To-Ground IoMT Network

FDRL Approach for Association and Resource Allocation in Multi-UAV Air-To-Ground IoMT Network

Abegaz Mohammed*,Aiman Erbad*, Hayla Nahom †, Abdullatif Albaseer*, Mohammed Abdallah*, and Mohsen Guizani‡
*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
‡Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE
Email: *{mabegaz,AErbad,amalbaseer, moabdallah}@hbku.edu.qa, †nahomh185@gmail.com, ‡mguizani@ieee.org

***Abstract*—In 6G networks, unmanned aerial vehicles (UAVs) can serve as aerial flying base stations (AFBS) with aerial mobile edge computing (AMEC) server capabilities. AFBS is an increasingly popular solution for delivering time-sensitive applications, extending network coverage, and assisting ground base stations in the healthcare systems for remote areas with limited infrastructure. Furthermore, the UAVs are deployed in the healthcare system to support the Internet of medical things (IoMT) devices in data collection, medical equipment distribution, and providing smart services. However, ensuring the privacy and security of patients' data with the limited UAV resources is a major challenge. In this paper, we present a federated deep reinforcement learning framework for resource allocation in UAV-enabled healthcare systems, where IoMT devices send their trained model parameters without transmitting sensitive raw data to the AMEC server. In the proposed framework, the IoMT device is associated with AFBS based on the quality of the data and its demand in order to maximize learning efficiency and accuracy. This work aims to minimize the computation costs of the IoMT devices while considering UAV resources and the fairness of UAV coverage. Simulation results prove that our proposed algorithm outperforms other baseline algorithms in learning accuracy and computational cost.**

***Index Terms*—IoMT, UAV, Privacy preservation, Computation cost, Federated learning, Resource allocation**

I. Introduction

The emergence of sixth-generation (6G) technologies offers a promising paradigm for intelligent Internet of medical things (IoMT) networks or intelligent Internet of Healthcare Things (IIoHT) in healthcare 4.0. The IoMT is an emerging technology that enables medical IoT devices and people to be incorporated into the healthcare industry to exchange healthcare data via a wireless connection [1]. In addition, IoMT increases reliability, provides the end-to-end services, reduce costs, and provides better services for society [1], [2].

However, resource constraints, privacy, network congestion, and delay are critical issues that must be addressed in IoMT to satisfy the quality of the service (QoS) and save patients' lives. These issues can affect the end-to-end delay and performance of data delivery in the healthcare system. The mobile edge computing (MEC) has been utilized to reduce delays in emergency packet delivery and reduce congestion in healthcare systems [1], [3]. However, the base station (BS) equipped with MEC services may fail due to artificial or natural calamities, or the MEC servers may become overwhelmed when dealing with ultra-dense IoMT/end-user devices. Furthermore, the conventional MEC network cannot fully satisfy the healthcare system's demands because most IoMT devices are mobile. In this regard, the Unmanned Aerial Vehicle (UAV) is used to enhance network coverage, relaying edge devices data to the central MEC server, compute tasks and allocate resources to devices, etc. In the healthcare industry, UAVs are used to collect and transfer medical data from IoMT devices to MEC servers, transfer medical data to desired patients and physicians, provide medical treatments and diagnoses to patients at any time and allocate different resources to IoMT devices [4], [5]. Therefore, the UAV-enabled healthcare system can alleviate the difficulties of personal and patient health monitoring and control in remote areas and during pandemics, epidemics, and emergencies.

Furthermore, machine learning (ML) methods have been applied in the healthcare industry to provide smart health services, optimizing system parameters, monitoring populations, and controlling chronic diseases [6]. In particular, the reinforcement learning (RL) and deep RL (DRL) methods are used in the healthcare industry to maximize energy efficiency, minimize communication latency, and allocate efficient resources [7] [8]. Federated learning (FL) is a recent ML paradigm that allows heterogeneous edge nodes to train data models and perform aggregation centrally, protecting data privacy. In [9], [10], the authors proposed a framework that utilizes FL and edge computing to help healthcare systems deal with the constraints of resources and privacy preservation issues. Yang *et al*. [11] presented an FL-based UAV-enabled network to preserve the privacy of the data training model for the edge user devices. Elayan *et al*. [12] proposed a deep FL framework for healthcare data monitoring and analysis using IoT devices to ensure medical data privacy and support decentralization.

The research works mentioned above demonstrated that federated reinforcement learning (FRL) has been used in 5G and B5G enabled healthcare systems to preserve privacy while efficiently allocating resources. Therefore, to address the problems of conventional MEC and resource limitations of IoMT devices while preserving medical data privacy, we propose a federated DRL (FDRL) framework for a UAV-enabled healthcare system that enables distributed learning and allocates resources efficiently. The combination of FL and UAV technology is an essential piece of technology for the healthcare system because it delivers ultra-low latency and addresses various ground IoMT needs. The main contributions of this paper are summarised as follows *(1)* We propose an FDRL framework for UAV-enabled healthcare systems that allows IoMT devices to share, aggregate, and update the DRL model parameters in a distributed manner. *(2)* We formulate association and resource allocation problems, then transform the problem to a Markov decision process (MDP) model and solve it using the DRL algorithm, a deep deterministic policy gradient (DDPG) to control a dynamic and high-dimensional network environment. *(3)* We conducted an extensive simulation using a real-world heartbeat datasets to prove that the proposed FDRL algorithm can achieve the theoretical analysis and outperform the existing benchmarks.

II. System Model

As shown in Fig. 1, we consider a UAV-enabled healthcare system consisting of multi-UAV deployments swarming over the ground network in a smart city. The UAVs are equipped with aerial MEC (AMEC) servers to provide services/resources to the IoMT devices (IMD). Moreover, one central server/SDN manages the air-to-ground (ATG) network. There is a set of IMDs denoted by $\mathcal{M} = \{1, \ldots, M\}$ and a set of UAVs in a clustered network denoted by $\mathcal{J} = \{1, \ldots, I\}$ and controlled by one UAV cluster head (UCH) u. In particular, the ground BS or small BS (SBS) may be overburdened as a result of providing a large number of IoT devices/live-stream events, BS/SBS may fail, and being unable to cover vast geographical areas such as remote areas and mobile devices. Therefore, UAV technology is a prominent solution to address these issues in B5G/6G network infrastructures. Without loss of generality, we use a 3D Cartesian coordinate system to express the position of IMD m as $(x_m, y_m, 0)$ and UCH u as (X_u, Y_u, H_u) at time slot t, where H_u is represents altitude of UCH. The horizontal distance between IMD m and UCH u can be written as:

$$D_{mu}(t) = \sqrt{(X_u - x_m)^2 + (Y_u - y_m)^2}, \forall m \in \mathcal{M}. \quad (1)$$

Let $\alpha^w_{mu}(t) \in \{0, 1\}$ be a binary variable representing IMD m association with UCH u and $\mathcal{A} = \{\alpha^w_{mu}(t)\}$ is the association matrix, $\forall m \in \mathcal{M}$. When $\alpha^w_{mu}(t) = 1$, the IMD m can share the model parameters after performing local training with UCH and the UCH u coverage should be i.e, $D_{mu}(t) \leq H_u tan\theta_u$; otherwise $\alpha^w_{mu}(t) = 0$ then IMD m computes the task locally. The ATG communication link between the UCH u and the m-th IMD with a specific probability can be

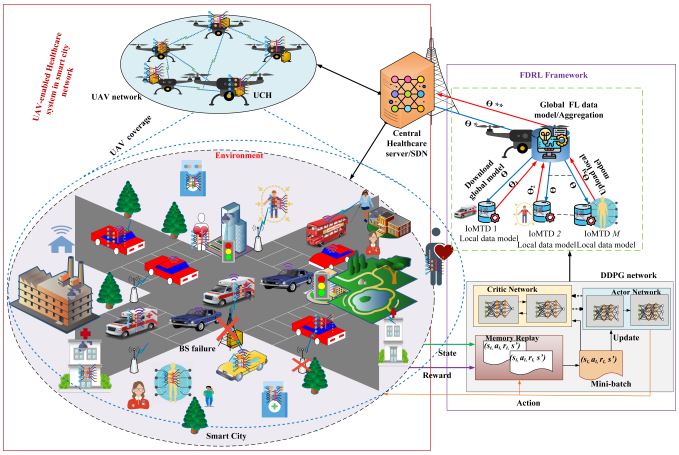


Fig. 1: Proposed system model.

modeled by path loss which includes the line of sight (LoS) and non-LoS (NLoS) [13]. The probability of LoS depends on the environment, the altitude, angle of elevation, and location of both the UCH and the IMD. Therefore, according to [13], [14] the LoS connection probability between UCH u and IMD m at time slot t is calculated as:

$$P^{LoS}_{mu}(t) = \frac{1}{1 + \varsigma_1 exp\big(-\varsigma_2(\frac{180}{\pi}\theta_{mu} - \varsigma_1)\big)}, \quad (2)$$

where ς_1 and ς_2 are constants depending on environment and θ_{mu} denotes the angle of elevation between UCH u and IMD m, which is given as $\theta_{mu} = tan^{-1}(\frac{H_u}{d_{mu}(t)})$. The transmission data rate of IMD m can be calculated as:

$$R_{mu}(t) = b_{mu}(t) log\left(1 + \frac{P_{mu}(t)h_{mu}(t)}{\sigma^2}\right), \quad (3)$$

where $b_{mu}(t)$ and $P_{mu}(t)$ denote the allocated bandwidth from UCH u to IMD m and the transmission power of IMD m to offload data to UCH u, respectively. σ^2 is noise power and $h_{mu}(t) = 10^{-Pl^{LoS}_{mu}/10}$ is channel gain between UCH u and IMD m at time slot t. $b_{mu}(t) \in [0, 1]$ is fraction of radio spectrum allocated to IMD. The allocation of spectrum resources should satisfy the following constraint:

$$\sum_{m \in \mathcal{M}} b_{mu}(t) \leq 1, \forall m \in \mathcal{M}. \quad (4)$$

We assume that the ATG system employs multiple channel access techniques based on the orthogonal frequency-division multiple access (OFDMA) approach [14]. The set of available subchannels is defined for each computing node as $w \in \mathcal{W} = \{1, 2, 3, ..., W\}$. Let B be an operational frequency band that can be divided into an equal subchannel $b = \frac{B}{W}$ [Hz], and assigned to IMDs using multiple radio access techniques. The IMD m then offloads tasks to the UCH at time slot t [15]. The allocated radio resource between IMD m and UCH u should meet the following constraints:

$$\sum_{m=1}^{M} \alpha^w_{mu}(t) b_{mu}(t) \leq B, \forall m \in \mathcal{M}. \quad (5)$$

Each computational node can serve at most m IMDs at time slot t.

A. Computing Model

Each IMD generates or collects sensitive data or time-tolerable data from patients or users, denoted as $N_m = (C_m, D_m, T_m, \beta_m)$, where C_m denotes the required number of CPU cycle to compute the data, D_m denotes the size of data transmitted or in transmission to edge node, T_m is the latency constraint and $\beta_m \in \{0, 1\}$ denotes data type is sensitive/critical or not. If $\beta_m = 1$, indicates sensitive/critical data and if $\beta_m = 0$, it is time tolerable data.

When the IMD m is not associated with UCH, it can compute tasks locally. The local computation time of the task is calculated as $T_m^{loc}(t) = \frac{D_m(t)}{f_m(t)} \leq T_m(t), \forall m \in \mathcal{M}$, where $f_m(t)$ is computation capacity of IMD m and should satisfy $0 \leq f_m(t) \leq f_m^{max}$. The energy consumption of local execution is calculated as: $P_m^{exe}(t) = \kappa_m(f_m^3)T_m^{loc}(t)$, where κ_m is the energy coefficient of CPU. The power constraint of IMD m is calculated by:

$$\sum_{t=1}^{T} \Big(P_m^{exe}(t) + P_{mu}(t) \Big) \leq P_m^{max}, \forall m \in \mathcal{M}, \tag{6}$$

where $P_{mu}(t)$ is transmission power consumption from IMD m into UCH, and P_m^{max} is maximum power budget of IMD m. When the IDM m is associated with UCH u at time slot t and offloads data into UCH, the data transmission latency is calculated as $T_{mu}^{tr}(t) = \alpha_{mu}^{w}(t)\frac{C_m(t)}{R_{mu}(t)}, \forall m \in \mathcal{M}$. The execution latency of offloaded data can be written as $T_{mu}^{exe}(t) = \alpha_{mu}^{w}(t)\frac{D_m(t)}{f_{mu}(t)}, \forall m \in \mathcal{M}$, where $f_{mu}(t)$ is allocated computation capacity from UCH u at time slot t. The total latency to process medical data is calculated as $T_{mu}(t) = T_m^{loc}(t)$, if $\alpha_{mu}^{w}(t) = 0$; otherwise $T_{mu}(t) = T_{mu}^{tr}(t) + T_{mu}^{exe}(t)$. The computation constraint of UCH u can be written as:

$$\sum_{m=1}^{M} f_{mu}(t) \leq F_u^{max}, \forall m \in \mathcal{M}, \tag{7}$$

where F_u^{max} is the maximum computation capacity of UCH u including UAVs.

B. Energy consumption Model

To compute the IMD m task, the energy consumption of UCH u includes transmission power, execution power, hovering power, and flying power consumption. We assume that each IMD m and UCH u adopts discrete power control [13]. The energy consumptionof UCH u for transmitting IMD m task at time slot t can be written as $E_{mu}^{tr}(t) = \alpha_{mu}^{w}(t)p_{mu}(t), \forall m \in \mathcal{M}$, where $p_{mu}(t)$ is the allocated transmission power of UCH u to IMD m. Further, the energy consumption for computing IMD m task at time slot t can be written as $E_{mu}^{ex}(t) = \hat{\kappa}_u f_{mu}^3 T_{mu}^{ex}(t), \forall m \in \mathcal{M}$, where $\hat{\kappa}_u$ effective capacitance coefficients of UCH u that depend on the CPU. The UCH u transmission power resource constraints can be written as:

$$\sum_{i=1}^{I} P_{mu}(t) \leq P_u^{max}, \forall j, \forall m \in \mathcal{M}, \tag{8}$$

where P_u^{max} is the maximum available transmission power resource block of UCH u. Therefore, the overall energy consumption to accomplish IMD m medical tasks can be written as $E_{mu}(t) = E_m^{IMD}(t)$ if $\alpha_{mu}^{w}(t) = 0$, otherwise $E_{mu}(t) = E_{mu}^{tr}(t) + E_{mu}^{ex}(t)$.

When the IMD is associated with the UCH u according to the current policy and other IMD information, the energy consumption is also calculated based on UCH u constraints such as flying, hovering, and execution energy consumption at the time slot t [14]. The flying energy consumption depends on flying distance and speed of UCH $V_u(t)$. Then, the flying energy consumption $E_u^{fl}(t)$ at time slot t is calculated as: $E_u^{fl}(t) = p_u^{fl}(t)\big(\frac{\sqrt{q_u(t)}}{V_u(t)}\big)$, where $p_u^{fl}(t)$ is flying power consumption of UCH u at time slot t and $V_u(t)$ represent speed of UCH u and $q_u(t) = [x_u(t) - x_u(t-1)]^2 + [y_u(t) - y(t-1)]^2$. The UCH u hovering energy consumption $E_u^{ho}(t)$ at time slot t is calculated as $E_u^{ho}(t) = \frac{p_u^{ho}(t)C_m(t)}{R_{mu}(t)}$, where $p_u^{ho}(t)$ denote the energy consumption during hovering at the time slot t. Therefore, the total energy consumption of UCH u in the time slot t is calculated as $E_u^{tot}(t) = \bar{\zeta}E_u^{fl}(t) + \hat{\zeta}E_u^{ho}(t) + \check{\zeta}E_u^{exe}(t)$, where $\bar{\zeta}, \hat{\zeta}, \check{\zeta}$ weights of the UCH u flight, hovering and computing energy consumption, respectively. In this study, our first aim is to allocate resources efficiently to IMDs while minimizing the energy consumption and latency of IMDs via optimizing both the offloading/association and resource allocation decisions. But, this may lead to an unfair process since one UCH may serve more IMDs than others. To address this, we adopt the fairness coverage of UCHs defined in [14]. It defines the level of fairness among UCHs that serve IMDs and among IMDs in the healthcare systems.

We formulate the optimization problem (9) to minimize the costs including energy consumption, delay, and resource allocation, depending on decision profile $\alpha_{mu}^{w}(t), b_{mu}(t)$ and the resources of UCH. The overall computation costs expressed as $\mathcal{Z}_{mu}(t) = \omega_t \alpha_{mu}^{w}(t)\big(\sum_{m=1}^{M} T_{mu}(t)\big) + \omega_e \alpha_{mu}^{w}(t)\big(\sum_{m=1}^{M} E_{mu}(t)\big)$, where ω_t and ω_e denote the weight of latency and energy consumption, respectively and $\omega_1 + \omega_2 = 1$.

$$\boldsymbol{P_1} \quad \min_{\mathcal{A},\mathcal{B},\mathcal{F},\mathcal{P}} \mathcal{Z}_{mu}(t) \tag{9}$$

$$\begin{aligned}
\textbf{S.t:} \quad & C1: \alpha_{mu}^{w}(t) \in \{0,1\}, \beta_m(t) \in \{0,1\}, \\
& \quad b_{mu}(t) \in [0,1], \forall m \in \mathcal{M}. \\
& C2: \sum_{u} \alpha_{mu}^{w}(t) = 1, \quad \sum_{u} b_{mu}(t) \leq 1. \\
& C3: \sum_{m=1}^{M} f_{mu}(t) \leq F_u^{max}, \forall m \in \mathcal{M}. \\
& C4: \sum_{m=1}^{M} p_{um}(t) \leq P_u^{max}, \forall m \in \mathcal{M}. \\
& C5: \sum_{m=1}^{M} \alpha_{mu}^{w}(t) b_{mu}(t) \leq B, \forall m \in \mathcal{M},
\end{aligned}$$

where $\big(\mathcal{A} = \{\alpha_{mu}^{w}(t)\}, \mathcal{B} = \{b_{mu}(t)\}, \mathcal{F} = \{f_{mu}(t)\}, \mathcal{P} = \{p_{mu}(t)\}, \forall m \in \mathcal{M}\big)$. Here, constraint (C1) indicates

the binary offloading decision strategy, data type of IMD m and radio resource, respectively. Constraints (C2) indicates that computation task compute at one place at time slot t, the allocated radio resource at time slot t should be equal to or less than 1. Constraints (C3, C4, C5) denote the maximum computation capacity, maximum transmission power, and maximum radio resource of computational node u, respectively.

C. Federated learning model

The classic DRL approaches lack training data and have a high overhead, making it difficult for agents to develop an accurate DRL model for individual learning. To tackle these issues, FL can be used to improve the local DRL model's training performance without relying on centralized training data. We utilize FL in a UAV-enabled healthcare system to improve decentralized resource allocation and association while maintaining the privacy of IMD data. The IMDs upload their local models to the AMEC server/SDN using optimal resource allocation and association policies. The AMEC server uses federated averaging to aggregate all local models into a global model. Then, the AMEC server sends the updated global model to the associated IMDs, and the IMDs train their local model based on the updated global model. For each IMD task, let $\mathcal{D}_m$ be a dataset of local IMD, and local training sample size is $\mathcal{D} = \{D_1, D_2, \ldots, D_M\}$. $\{\mathcal{D}_{mu}\}$ is a dataset on UCH to train the global model. For the m-th IMD, the sum loss function on dataset $\mathcal{D}_m$ can calculated as:

$$F_m(\boldsymbol{\theta}_u) = \frac{1}{\sum_{m\in\mathcal{M}}} \sum_{m=1}^{M} F(\boldsymbol{\theta}_u). \tag{10}$$

Each associated IMD m trains its model parameters $\boldsymbol{\theta}_m(t)$ based on its dataset $\mathcal{D}_m$ by calculating the local stochastic gradient descent $\bigtriangledown F_m(\boldsymbol{\theta}_u(t))$

$$\boldsymbol{\theta}_m(t)) = \boldsymbol{\theta}_u(t)) - \phi \bigtriangledown F_m(\boldsymbol{\theta}_u(t)), \tag{11}$$

where $\phi > 0$ is the learning step. The AMEC server performs global model aggregation by averaging and updating global model parameters expressed as:

$$\boldsymbol{\theta}_u(t+1) = \frac{1}{\sum_{m\in\mathcal{M}} D_m} \sum_{m}^{M} \boldsymbol{\theta}_m(t+1). \tag{12}$$

Algorithm 2 depicts the proposed FDRL algorithm process.

III. FDRL-based Solution

In this section, we present an FDRL-based privacy-preserving resource allocation and association policy to solve the resource allocation and association problems in (9). We incorporate the DRL (DDPG) with the FL. The problem with the ATG network is that the environment is dynamic, and the action space is continuous. The DDPG algorithm [16] is used to solve this problem. It can make an optimal association and resource allocation decision based on IMD demand information, network coverage, and resource block of UCH. However, the IMDs transmit the raw data to the edge or central server, which trains the learning model and obtains the optimal policy. It may transmit incorrect data or requests, as medical data is sensitive and may be compromised during the training process, and raw data transmission consumes resources. Therefore, we use FL, which allows each IMD to train a local model and then transmit the parameters of the locally trained model to acquire a global model by aggregating the local model at UCH or SDN. It minimizes the global model's loss function and computation costs in terms of communication latency and energy consumption while maintaining the privacy of the IMD data.

Algorithm 1 DDPG-based solution

1: Randomly initialize **critic's network** $Q(s, a|\theta^Q)$ and **actor's network** $\mu(s_t|\theta^\mu)$ with weight θ^Q and θ^μ
2: Initialize actor's and critic's target networks $Q'(.)$ and $\mu'(.)$, with weights $\theta^{Q'} \leftarrow \theta^Q$ and $\theta^{\mu\prime} \leftarrow \theta^\mu$
3: Initialize the memory replay $\mathcal{B}$
4: **for** episode = $[1, 2, \ldots, 1000]$ **do**
5: Initialize ATG environment
6: Receive an initial state $s(0)$
7: **for** time step : [t = 1, 2,...,T] **do**
8: Based on the policy μ, select action $a(t) = \mu(s(t)) + \varsigma$, ς is exploration noise
9: **if** $\psi_u(t) > 0 \cap \nu_u(t) > 0$ **then**
10: Execute action $a(t)$ and obtain the immediate reward $r(t)$.
11: Observe the next state $s(t+1)$.
12: Collect and store transition tuples $(s(t), a(t), r(t), s(t+1))$ into memory replay $\mathcal{B}$.
13: Randomly sample mini-batch $\mathcal{H}$ of transition tuples from $\mathcal{B}$.
14: Update the **critic main-network** $Q(s, a|\theta^Q)$ and Update the **actor main-network** $\mu(s(t)|\theta^\mu)$ by policy gradient and loss function.
15: Update **actor's target network and critic target-network by** $\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$
16: **end if**
17: **end for**
18: **end for**

A. Deep Reinforcement Learning Framework

First, we transform the problem (9) into a DRL basic idea of an MDP model [14], [17]. The AMEC server acts as a learning agent. Define the MDP tuples as follow:

State Space $S(t)$: The state space is a set of states $s(t) \in S(t) = \{s(t)\}$ in the environment, i.e., $s(t) = \{\beta_m(t), \psi_u(t), T_m(t), \nu_u(t)\}$, where $\beta_m(t), \psi_u(t), T_m(t)$ and $\nu_u(t)$ denote task type of IMD, UCH resource blocks, maximum time delay of task N_m and UCH coverage range at time slot t, respectively.

Action space $A(t)$: The agent selects association and resource allocation actions $a(t)$ from action space $A(t)$ depending on the observed state and the current policy π, where $\pi : S \rightarrow A$. $a(t) = \{\alpha^w_{mu}(t), b_{mu}(t), f_{mu}(t), p_{mu}(t)\}$.

Reward function $R(t)$: The reward function is the objective function of (9). The agent obtains minimal computation costs, including communication latency and energy consumption, after completing the IMD task N_m at time slot t. The reward function is defined as $r(t) = -(\omega^t r^t + \omega^e r^e)$. Then the overall reward function is given as $R(t) = \sum_{t=1}^{T} r(t)$. The objective of the agent is to maximize the long-term reward, which is given as: $\pi^* = \text{argmax}_a \, \mathbb{E}_\pi[\sum_{t=1}^{+\infty} \gamma^{t-1} r(t)]$, where γ is discount factor on immediate reward $r(t)$.

Algorithm 2 FDRL-based association and resource allocation

1: **Inputs:** Execute resource allocation and association by running **Algorithm 1**
2: Initialize: global parameters $\boldsymbol{\theta}_u(0)$ at AMEC server, number of communication round T, and learning step ϕ
3: **for** time slot = $1, 2, ..., T$ **do**
4: **for** Each IMD m **do**
5: Compute its local update.
6: Train the local model parameter.
7: Return $\boldsymbol{\theta}_m(t+1)$
8: **end for**
9: Upload $\boldsymbol{\theta}_m(t+1)$ to the AMEC server.
10: AMEC server receive $\boldsymbol{\theta}_m(t+1)$ from each IMD m.
11: Update the global model on AMEC server by Equ. (12).
12: **end for**
13: **Output:** Obtain final global FL model parameter $\boldsymbol{\theta}_u$.

1) DDPG based solution: The DDPG algorithm can discretize state and action spaces to deal with continuous state and action spaces. It solves the problem by combining the advantages of deep neural network (DQN) with a policy gradient. It is a hybrid network comprised of a critic and an actor (make action decision); both make use of a DQN. The critic evaluates each state-action pair using a Q-function [16]. As any DRL algorithm, DDPG algorithm has training and testing phases. Regarding the correlation between transitions utilized in the training stage, experience replay technology is used in DDPG to minimize the convergence rate. Specifically, storing O_t transitions in a replay memory buffer and randomly sampling a mini-batch of transitions from the buffer to train the DDPG model, i.e., updating the actor and critic's parameters until they converge. The actor-network and critic network parameters are updated according to the policy gradient and loss function [16], respectively. In particular, the parameter matrix of the actor's evaluation network (θ^μ) is updated by applying the chain rule [16] as $\bigtriangledown_{\theta^\mu} J(\theta^\mu)$, where $\bigtriangledown_{\theta^\mu}$ is gradient of θ^μ and $J(\theta^\mu)$ is the policy objective function. The critic network modifies the parameters of its evaluation network (θ^Q) the gradient of ($\bigtriangledown_{\theta L(\theta^Q)}$) to minimize the loss function, $L(\theta^Q) = \mathbb{E}\Big((Q(s,a)|_t - (r(t) + \gamma Q'(s,a)|_{t+1})\Big)^2$, where $Q'(s,a)$ Q-function of the critic's target network. Then the agent uses soft updating technique to update the parameters of target networks as $\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$, where τ is the soft update coefficient. The DDPG-based association with the resource allocation process is shown in Algorithm 1.

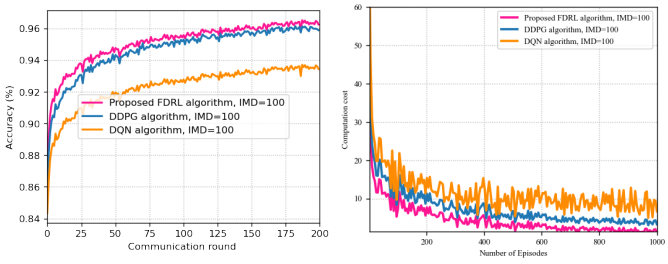
IV. Simulation Results

To demonstrate the performance of the FDRL-based association and resource allocation framework through extensive simulation. For comparison, we use DDPG and the DQN algorithms and utilize a real-world heartbeat dataset to evaluate the performance of our proposed FDRL algorithm. This dataset [18] includes a group of heartbeat signals, i.e., electrocardiogram (ECG) signals. These signals represent normal and abnormal cases. The deployment and parameter configuration of the ATG network depends on [14]. We consider M=100 IMD randomly distributed in 500m×500m. The UAV network is swarming over the ground network and hovering at 100m. The system bandwidth and subchannel bandwidth are 20MHz and 80KHz, respectively.

TABLE I: Simulation Parameters

Parameters	**Values**
Number of SDN,UCH and UAVs	1,1,5
Path loss exponent	2.2
P_m^{max} and P_u^{max}	10dBm and 30dBm
f_m^{max} and f_u^{max}	2.0 GHz, 15GHz
$\hat{\kappa}_m$, $\hat{\kappa}_u$	10^{-27} , 10^{-28}
Velocity of UCH	25m/s
Average power gain and noise power	-60dBm, -110dBm
Discount factor	0.99
Number of episodes	1000
Steps per episodes	200
Learning rate of actor and critic	10^{-3}, 10^{-2}
Soft update	0.001

The data size of IMD is [100,1000] KB, and the requested CPU cycle is [0.5,1.5] GHz. The size of the replay memory buffer is $2.5X10^6$; the mini-batch size is 1024, and the discount factor $\gamma = 0.9$. Table 1 summarizes other simulation parameters.

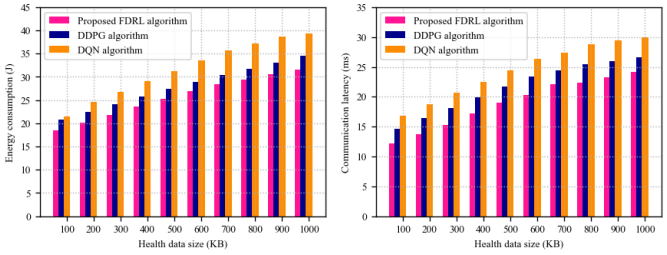


(a) Accuracy rate. (b) Communication cost.

Fig. 2: Communication analysis.

As shown in Fig. 2a, the accuracy of the proposed scheme and benchmark algorithms are evaluated using a heartbeat dataset. We have taken the better client/IMD participation rate of 0.5 from trial and error in the FL training. With increasing rounds of communication and global updates, the accuracy of all algorithms rapidly increases in the first 15 rounds and gradually converges after 25 rounds. The evaluation results demonstrate that the accuracy rate is affected by the quality

of training data, communication rounds, and the number of clients/IMDs.



(a) Energy consumption. (b) Communication latency.

Fig. 3: Performance analysis based on health data size.

The simulation results show that the FDRL and DDPG algorithms outperformed the DQN algorithm in terms of accuracy. However, the DDPG algorithm has a lower accuracy rate than the FDRL algorithm. As shown in Fig. 2b, the system cost gradually declines with increasing learning episodes in all algorithms. Initially, the system cost of all algorithms is high due to less learning experience in the high dimension of state space and action space. The proposed FDRL algorithm has a lower system cost than baseline algorithms, which is a significant advantage for minimizing the latency and energy consumption of the local IMDs and UAVs. The DQN algorithm has a higher system cost than the other two algorithms because it cannot handle continuous action space in dynamic multi-UAV-enabled IoMT networks. The FL model synchronizes the local and global models with low communication and energy consumption. In general, the proposed FDRL algorithm outperforms the baseline algorithm to minimize system costs.

Fig. 3 shows the impact of medical data size on energy consumption and latency costs; as data size increases, system costs gradually increase. In Fig. 3a, we can observe that the energy consumption increases with the data size of tasks. It means that more data can be offloaded and computed; it also has an impact on system performance and task completion deadlines. The proposed FDRL algorithm outperforms the baseline algorithms in terms of energy consumption, reducing 8.953% and 20.625% when compared to the DDPG and DQN algorithms, respectively. The increasing data size of tasks affects system latency due to increased transmission and execution time, as shown in Fig.3b. The AMEC server allocates more resource blocks to accomplish IMD tasks within the deadline. When compared to the DDPG and DQN algorithms, the proposed FDRL algorithm reduces latency by 11.75% and 19.805%, respectively.

V. CONCLUSION

In this paper, we proposed an FDRL framework for privacy-preserving and secured resource allocation and IMD association problems in the UAV-enabled healthcare system. Due to various optimization variables in the environment, the problems are difficult to solve; thus, we transform them into an MDP model. Then, the DRL algorithm was integrated with the FL method to minimize the edge server burden/traffic and ensure medical data privacy. The DDPG algorithm was used to optimize these problems and control the continuous action space. The numerical results have proven that the proposed method outperforms the benchmark algorithm in terms of accuracy and computation costs.

REFERENCES

[1] I. Ud Din, A. Almogren, M. Guizani, and M. Zuair, "A decade of internet of things: Analysis in the light of healthcare applications," *IEEE Access*, vol. 7, pp. 89 967–89 979, 2019.

[2] F. Al-Turjman, M. H. Nawaz, and U. D. Ulusar, "Intelligence in the internet of medical things era: A systematic review of current and future trends," *Computer Communications*, vol. 150, pp. 644–660, 2020.

[3] Z. Ning, P. Dong, X. Wang, X. Hu, L. Guo, B. Hu, Y. Guo, T. Qiu, and R. Y. K. Kwok, "Mobile edge computing enabled 5g health monitoring for internet of medical things: A decentralized game theoretic approach," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 463–478, 2021.

[4] S. Ullah, K.-I. Kim, K. H. Kim, M. Imran, P. Khan, E. Tovar, and F. Ali, "Uav-enabled healthcare architecture: Issues and challenges," *Future Generation Computer Systems*, vol. 97, pp. 425–432, 2019.

[5] S. Aggarwal, N. Kumar, M. Alhussein, and G. Muhammad, "Blockchain-based uav path planning for healthcare 4.0: Current challenges and the way ahead," *IEEE Network*, vol. 35, no. 1, pp. 20–29, 2021.

[6] C. Yu, J. Liu, and S. Nemati, "Reinforcement learning in healthcare: A survey," *arXiv preprint arXiv:1908.08796*, 2019.

[7] E. Baccour, N. Mhaisen, A. A. Abdellatif, A. Erbad, A. Mohamed, M. Hamdi, and M. Guizani, "Pervasive ai for iot applications: Resource-efficient distributed artificial intelligence," *arXiv preprint arXiv:2105.01798*, 2021.

[8] A. A. Abdellatif, N. Mhaisen, Z. Chkirbene, A. Mohamed, A. Erbad, and M. Guizani, "Reinforcement learning for intelligent healthcare systems: A comprehensive survey," *arXiv preprint arXiv:2108.04087*, 2021.

[9] Z. Xue, P. Zhou, Z. Xu, X. Wang, Y. Xie, X. Ding, and S. Wen, "A resource-constrained and privacy-preserving edge-computing-enabled clinical decision system: A federated reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9122–9138, 2021.

[10] W. Y. B. Lim, S. Garg, Z. Xiong, D. Niyato, C. Leung, C. Miao, and M. Guizani, "Dynamic contract design for federated learning in smart healthcare applications," *IEEE Internet of Things Journal*, 2020.

[11] H. Yang, J. Zhao, Z. Xiong, K.-Y. Lam, S. Sun, and L. Xiao, "Privacy-preserving federated learning for uav-enabled networks: Learning-based joint scheduling and resource management," *IEEE Journal on Selected Areas in Communications*, 2021.

[12] H. Elayan, M. Aloqaily, and M. Guizani, "Sustainability of healthcare data analysis iot-based systems using deep federated learning," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[13] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal lap altitude for maximum coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.

[14] 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*, pp. 1–1, 2021.

[15] X. Qin, Z. Song, Y. Hao, and X. Sun, "Joint resource allocation and trajectory optimization for multi-uav-assisted multi-access mobile edge computing," *IEEE Wireless Communications Letters*, vol. 10, no. 7, pp. 1400–1404, 2021.

[16] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[17] 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.

[18] S. Fazeli, "Ecg heartbeat categorization dataset," May 2018. [Online]. Available: https://www.kaggle.com/shayanfazeli/heartbeat