Delay Minimization in Uplink UAV-aided FL Systems with MC-NOMA

Manh Cuong Ho, 이동현, Quang Tuan Do, Tung Son Do, 김동완*, 조성래
{hmcuong, dhlee, dqtuan, tsdo}@uclab.re.kr, dongwankim@dau.ac.kr, srcho@cau.ac.kr

Chung-Ang Univ., *Dong-A Univ.

ABSTRACT

In the current, Federated learning (FL) plays an important role in future Internet of Things (IoT) networks as a promising decentralized machine learning technique. It ensures the security and privacy of user data while efficiently utilizing communication resources. This paper tackles the challenge of minimizing system delay in UAV-aided FL systems with multi-carrier non-orthogonal multiple access (MC-NOMA). We jointly optimize UAV location, power allocation, and sub-channel association variable allocation to achieve this goal. We transformed the proposed optimization problem into a Markov decision process (MDP) and applied a reinforcement learning algorithm called deep deterministic policy gradient (DDPG) to solve the MDP problem.

I. INTRODUCTION

In UAV-assisted FL networks, user devices collaboratively perform a learning task by uploading their local learning models to an UAV without the need to share all their training data. However, the limited availability of wireless resources, the limited energy of devices, and the movement of UAV or users are main challenges that can negatively impact FL performance. Therefore, prioritizing delay and energy reduction is crucial for optimizing FL implementation to overcome the limitations of UAV networks. In this work, we focus on minimizing uploading time in UAV-aided FL systems using MC-NOMA by optimizing UAV location, power allocation, and sub-channel association variable allocation. The structure of this paper is as follows: Section II describes the system model and problem formulation. Section III explains the deep deterministic policy gradient (DDPG) algorithm used to optimize system delay. Finally, Section IV concludes this work.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider an UAV-based FL network where a single antenna UAV serves N single antenna users. The UAV can operate as an aerial server by carrying a cloud server and hover at a fixed point in the aerial. We assume UAV and users have sufficient batteries to complete whole FL training process. Each user device is equipped with an on-board computing processor to train the local model. The system's operation T is divided into time slots, each time slot is indexed by t = {1, 2,..., T}. At the beginning of time slot t, the server broadcasts the global parameters to all users. Each user n uses its own data to train a local model. Then, each user n sends its local parameters to the UAV in the uplink. After receiving the final local model, the server combines all the local models to create a new global model and broadcasts the updated global parameters to all users, initiating the next time slot. We assume that there is a dedicated control channel for the simple slot and UAV can feedback all SINR info to all user through the dedicated channel. Denote by $q = \{a, b\} \in R^{2\times 1}$ and $q_{n,t} = \{a_n, b_n\} \in R^{2\times 1}$ are the x, y coordinates of the UAV and user n. The distance $d_{n,t}$ between the UAV and user n at time slot t is defined as:

$$d_{n,t} = \sqrt{(H-h_n)^2 + ||q - q_{n,t}||_2^2},$$

where H, q, h_n, $q_{n,t}$ are the height of UAV, coordinate of the UAV, the height of user n, and the coordinate of the user n, respectively. We assume that the UAV moves with fixed altitude H within a circle of radius C to transmit signal to users. Let $g_{n,t}$ denote the channel coefficient between the UAV and user n at time slot t. In this paper, we assume that links between the UAV and users are NloS links. For user k in sub-channel m at time slot t, the achievable rate in bits per second (bps) is given as

$$r_{k,m,t} = \nabla_{k,m,t} b_t \log_2\left(1 + \frac{|g_{k,m,t}|^2 p_{k,t}}{\sum_{k'=1}^{k-1} \nabla_{k',m,t}|g_{k',m,t}|^2 p_{k',t} + \sigma^2 b_t}\right),$$

where $b_t, \nabla_{k,m,t}, p_{k,t}, \rho^2, g_{k,m,t}$ are the bandwidth of each sub-channel, the sub-channel association variable, the transmit power of user k, the power spectral density of the Gaussian noise at time slot t, and the channel gain between UAV and user k. The sub-channel association variable indicates user selection for each sub-channel. If user k selects the sub-channel m, we set the sub-channel association variable $\nabla_{k,m,t} = 1$. Otherwise, $\nabla_{k,m,t} = 0$.

The model size of user n is denoted as Z_n. Then the uploading time is calculated as

$$t_{k,m,t}^u = \frac{Z_n}{r_{k,m,t}}.$$

The uploading time of all users should be lower than the completion time of a FL round t^{to}. The system uploading time at time slot t can be stated as

$$t^u = max_{n\in N}\,(t_{n,t}^u),$$

where $t_{n,t}^u$ is the uploading time of user n.

We consider the optimization of location of UAV, power allocation, and sub-channel association variable allocation to minimize the uploading time of UAV-aided FL systems with MC-NOMA.

$$\boldsymbol{min_{q,p,\nabla}} \sum_{t=1}^{T}\left(\sum_{n=1}^{N} \boldsymbol{t^u}\right)$$

s.t.

$$0 \le p_{n,t} \le p_{n,t}^{max}, \forall n \in N, \forall t \in T$$
$$\sum_{m=1}^{M} \nabla_{n,m,t} \in \{1,0\}, \forall n \in N, \forall t \in T$$

$$||q||_2^2 = a^2 + b^2 \le C^2$$
$$q, q_{n,t} \ge 0, \forall n \in N, \forall t \in T$$
$$t^u \le t^{to}$$

III. DDPG-BASED SYSTEM DELAY EFFICIENT FL ALGORITHM

To address the problem, we reformulated the optimization challenge as a MDP and developed a DDPG-based reinforcement learning algorithm to effectively solve it. The MDP model is defined as a tuple (S, A, R), where S represents the set of states of the environment, A represents the set of actions, and R: S × A → R represents the reward function, indicating the reward the agent receives from the environment for taking action a in state s. Besides, an agent is set at the server. We clarify the definitions of the state space, action space, reward, and penalty as follows:

State space: At each time slot t, the agent observes the network environment and determines the system state at the beginning of the time slot. This state includes the location of the UAV, the location of the users, the available bandwidth, and the model size.

$$s = \{\boldsymbol{q}, \boldsymbol{q}_{n,t}, \boldsymbol{b}_t, \boldsymbol{Z}_{n,t}\}, \forall n \in N, \forall t \in T$$

Action: The action taken in response to the current state s_t includes adjusting the transmit power, optimizing the UAV location, and selecting the sub-channel association variable.

$$a = \{\boldsymbol{q}', \boldsymbol{p}_{n,t}, \boldsymbol{\nabla}_{n,t}\}, \forall n \in N, \forall t \in T$$

Reward: Since our goal is minimization of the uploading time, the decision reward function is

$$r_d = -t^{ct}$$

To satisfy the constraints, we set time penalty as

$$p_t = \begin{cases} 1, if\ t^u \ge t^{to} \\ 0, otherwise \end{cases}$$

Therefore, the total decision reward function of the model at the time section t_e is formulated as

$$r = r_d - p_t$$

We introduce a DDPG-based energy-efficient FL algorithm, outlined in Algorithm 1, as an effective solution to address our optimization problem. The key components of the DDPG algorithm include utilizing two primary deep neural networks (DNNs): a policy network and a critic network. Our goal is to determine the optimal policy, defined as

$$\mu^*(s|\theta^\mu) = argmax_{a \in A}\{Q(s_t, a_t)|\theta^Q\},$$

where Q is the value of chosen action a_t at state s_t. Thus, we find the optimal joint action a^* that maximinxes the expected cumulative Q-value Q^*. We achieve this solving the following Bellman equation:

$$Q^*(s_t, a_t)|\theta^Q = E\left[max_{a \in A}\left[r_t + \gamma Q^*(s_{t+1}, a_{t+1})|\theta^Q\right]\right].$$

The actor network, responsible for learning the policy, takes the current state as input and generates the corresponding action as output. Both the actor network and the critic network are updated using a policy gradient and a loss function, respectively.

The DDPG-based system delay algorithm is as follow

Algorithm 1: DDPG-based System Delay Efficient FL Algorithm

```
Initialize hyperparameters.
Initialize the actor network, μ(s|θ^μ), and the critic network, Q(s,a|θ^Q) with weights θ^μ and θ^Q.
Initialize the target actor network μ'(s|θ^μ') and the target critic network Q'(s,a|θ^Q') with weights θ^μ' ← θ^μ and
  θ^Q' ← θ^Q.
Initialize a replay buffer H.
for each episode e = 1, 2, ..., e_max do
    Initialize system state s_1 and random action nose χ_0.
    for t = 1, 2, , , , , T do.
        Observe s_t.
        Execute overall action a_t^*.
        Observe reward r_t and next state s_{t+1}.
        Store experience in the buffer H and uniform sample a batch of H to train the DNNs.
        Update the actor network and the critic network.
        Update the target actor network and the target critic network.
    end for
end for
```

IV. CONCLUSIONS

This paper focuses on minimizing system delay in UAV-assisted FL systems using MC-NOMA. Initially, we optimized UAV location, power allocation, and sub-channel association variable allocation jointly to minimize uploading time. Subsequently, we formulated this joint optimization problem into a Markov decision process (MDP). Then, we proposed a DDPG-based reinforcement learning algorithm to effectively solve the MDP and optimize system delay.

ACKNOWLEDGMENT

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support programs(IITP-2024-RS-2022-00156353 and IITP-2024-RS-2023-00258639) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation).

References

[1] Ho, M. C., Tran, A. T., Lee, D., Paek, J., Noh, W., & Cho, S. (2023). A DDPG-based energy efficient federated learning algorithm with SWIPT and MC-NOMA. ICT Express.

[2] Truong, T. P., Dao, N. N., & Cho, S. (2022). HAMEC-RSMA: Enhanced aerial computing systems with rate splitting multiple access. IEEE Access, 10, 52398-52409.

[3] Kwon, D., Jeon, J., Park, S., Kim, J., & Cho, S. (2020). Multiagent DDPG-based deep learning for smart ocean federated learning IoT networks. *IEEE Internet of Things Journal*, *7*(10), 9895-9903.