



**CISTER**

Research Centre in  
Real-Time & Embedded  
Computing Systems

# Journal Paper

---

## **Onboard Deep Deterministic Policy Gradients for Online Flight Resource Allocation of UAVs**

Early Access

**Kai Li\***

**Yousef Emami\***

**Wei Ni**

**Eduardo Tovar\***

**Zhu Han**

---

\*CISTER Research Centre

CISTER-TR-200602

2020/06/15

# Onboard Deep Deterministic Policy Gradients for Online Flight Resource Allocation of UAVs

Kai Li\*, Yousef Emami\*, Wei Ni, Eduardo Tovar\*, Zhu Han

\*CISTER Research Centre

Polytechnic Institute of Porto (ISEP P.Porto)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8321159

E-mail: kai@isep.ipp.pt, emami@isep.ipp.pt, Wei.Ni@data61.csiro.au, emt@isep.ipp.pt, zhan2@uh.edu

<https://www.cister-labs.pt>

## Abstract

In Unmanned Aerial Vehicle (UAV) enabled data collection, scheduling data transmissions of the ground nodes while controlling flight of the UAV, e.g., heading and velocity, is critical to reduce the data packet loss resulting from buffer overflows and channel fading. In this letter, a new online flight resource allocation scheme based on deep deterministic policy gradients (DDPG-FRAS) is studied to jointly optimize the flight control of the UAV and data collection scheduling along the trajectory in real time, thereby asymptotically minimizing the packet loss of the ground sensor networks. Numerical results confirm that the proposed DDPG-FRAS can gradually converge, while enlarging the buffer size can reduce the packet loss by 47.9%.

# Onboard Deep Deterministic Policy Gradients for Online Flight Resource Allocation of UAVs

Kai Li, *Senior Member, IEEE*, Yousef Emami, *Student Member, IEEE*, Wei Ni, *Senior Member, IEEE*, Eduardo Tovar, *Member, IEEE*, and Zhu Han *Fellow, IEEE*

**Abstract**—In Unmanned Aerial Vehicle (UAV) enabled data collection, scheduling data transmissions of the ground nodes while controlling flight of the UAV, e.g., heading and velocity, is critical to reduce the data packet loss resulting from buffer overflows and channel fading. In this letter, a new online flight resource allocation scheme based on deep deterministic policy gradients (DDPG-FRAS) is studied to jointly optimize the flight control of the UAV and data collection scheduling along the trajectory in real time, thereby asymptotically minimizing the packet loss of the ground sensor networks. Numerical results confirm that the proposed DDPG-FRAS can gradually converge, while enlarging the buffer size can reduce the packet loss by 47.9%.

**Index Terms**—Unmanned aerial vehicles, Flight control, Data collection, Deep reinforcement learning

## I. INTRODUCTION

For data collection in large sensor networks, Unmanned Aerial Vehicles (UAVs) can be employed to visit remote sensor nodes that are airlifted and deployed in remote areas, as shown in Figure 1. UAVs have also been studied to help mitigate the impact of COVID-19 outbreak [1]. Thanks to the high mobility and maneuverability, a UAV can move sufficiently close to each of the sensor nodes on the ground, to exploit short-distance line-of-sight (LoS) communications and collect their sensory data [2], [3]. Due to new materials and technologies, solar-powered UAVs have been developed to fly over a long distance without landing, e.g., Hawk 30 by SoftBank [4] and Morning Star by AVIC [5].

Data packets are generated and buffered at the sensor nodes, where the buffers have finite sizes. New data packets have to be dropped when the buffers overflow. The flight cruise of the UAV is controlled by adapting the headings and patrol velocities to visit specific ground nodes and collect their data before the buffers overflow. Moreover, the UAV's flight control results in changing channel conditions between the ground nodes and the UAV. Having a ground node transmit data when signal-to-noise ratio (SNR) is poor is likely to lead to packet reception errors at the UAV.

The ground sensor node can progressively harvest energy from multiple renewable energy sources, e.g., solar panel,

electret-based wind turbine, or wireless power transfer, to charge its battery. Because of time-varying and independent energy harvesting conditions at the nodes, the battery energy level of the ground nodes can substantially differ between each other. Some nodes can be fully charged while others are running out of energy. Scheduling the ground nodes with low energy levels to transmit can cause other fully charged nodes to experience buffer overflow [6]. In practice, the complete up-to-date information of data queue length, battery energy, and channel SNR of the ground nodes is not available at the UAV. Therefore, the scheduling of the ground nodes' data transmission is critical for onboard flight control of the UAV to prevent buffer overflow and packet loss caused by channel fading.

Some works in the literature have developed Markov Decision Process (MDP) for collision-free flight trajectory of the UAV, target tracking, or power allocation, e.g., [7], [8], and [9]. A resource allocation strategy was studied in [10] to reduce the packet loss in energy harvesting sensor networks, given the predetermined transmission probability and channel statistical information. Based on the statistical information of the MDP, the resource allocation in [10] was solved by dynamic programming. In [11] and [12], the scheduling of power transfer and data transmission was studied in the small-scale static wireless sensor network, where Q-learning was developed to extend network lifetime.

In this letter, we study a new onboard flight resource allocation of a UAV to minimize the overall data loss of many ground nodes. The flight resource allocation of the UAV is first formulated as an MDP by jointly considering the battery energy level, the data queue backlog, and the SNR state information of the ground nodes. The optimal flight resource allocation to the formulated MDP problem can be solved by Q-learning. However, Q-learning is known to suffer from the curse-of-dimensionality, which is impractical for the online flight resource allocation of the UAV.

Deep Q-Network (DQN) in [13] is applied to schedule energy harvesting and data transmission with an enlarged state and action space of the MDP, given a predetermined and fixed flight trajectory of the UAV. In contrast to [13], we jointly optimize the heading and the patrol velocity of the UAV, as well as the data collection schedule of the ground nodes. Particularly, the up-to-date knowledge of battery levels, queue lengths of the ground nodes and the SNR is not available at the UAV. For online control of the headings and velocities of the UAV, the onboard

K. Li, Y. Emami, and E. Tovar are with Real-Time and Embedded Computing Systems Research Centre (CISTER), 4249-015 Porto, Portugal (E-mail: {kai,emami,emt}@isep.ipp.pt).

W. Ni is with Commonwealth Scientific and Industrial Research Organization (CSIRO), Sydney, Australia (E-mail: wei.ni@data61.csiro.au).

Z. Han is with Electrical and Computer Engineering Department, University of Houston, Texas, US (E-mail: zhan2@uh.edu).

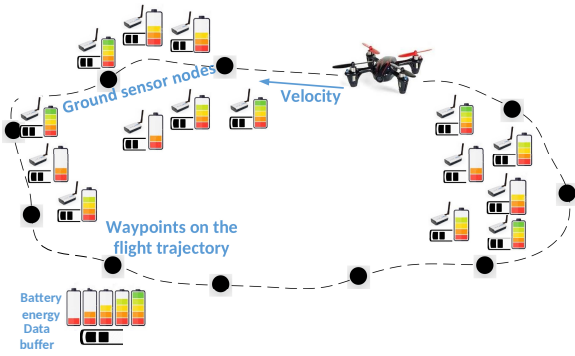


Fig. 1: Data packets are generated and buffered at the ground sensors, where the buffers have finite sizes. The ground sensor can progressively harvest energy from multiple renewable energy sources, e.g., solar panel, electret-based wind turbine, or wireless power transfer, to charge its battery. The UAV continuously adapts the headings and the patrol velocities to visit the ground sensors and collect their data before the buffers overflow.

flight resource allocation contains high dimensional and continuous (real valued) action spaces. Therefore, the deep Q-network based on discrete action spaces, such as [13], is not applicable to continuous flight control problem.

To minimize the packet loss of the whole system, we propose a new Deep Deterministic Policy Gradients based Flight Resource Allocation (DDPG-FRA) framework, where deep Q-learning is carried out in the continuous action space for the online flight control of the continuous headings and patrol velocities. DDPG-FRA optimally decides the ground node to be interrogated for data collection along the flight trajectory of the UAV. Moreover, DDPG-FRA avoids the curse-of-dimensionality, compared with MDP and Q-learning which requires the discretization of the state space.

The remaining part of this letter is organized as follows. System and network models are investigated in Section II. Section III studies the MDP formulation of the flight resource allocation problem. In Section IV, onboard flight resource allocation based on DDPG is proposed for the UAV-enabled data collection. Section V presents numerical results and performance evaluation of the proposed DDPG-FRA framework. Section VI concludes the letter.

## II. SYSTEM AND NETWORK MODELS

We consider that  $N$  ground sensor nodes are deployed in a remote area. Node  $i$  ( $i \in [1, N]$ ) can harvest renewable energy from the environment to charge its battery for powering its operations, e.g., sensing, computing and communication. The UAV that acts as a data mule or airborne base station patrols along the trajectory [14]. Let  $\zeta(t)$  denote the location of the UAV in space at time  $t$ . The instantaneous velocity of the UAV is  $v_{\zeta(t)}$ . The complex coefficient of the reciprocal wireless channel between the UAV and node  $i$  at  $t$  is  $h_i(\zeta(t))$  [15]. In particular,  $\zeta(t)$  and  $h_i(\zeta(t))$  take values in the continuous spaces. Moreover, the modulation-and-coding scheme of ground node  $i$ , denoted by  $\phi_i(t)$ , can be adapted for data transmission,

where  $\phi_i(t) \leq \Phi$ , and  $\Phi$  is the total number of modulation-and-coding schemes. Particularly, the typical modulations BPSK, QPSK, and 8PSK are denoted by  $\phi_i(t) = 1, 2$ , and 3, respectively, and  $2^{\phi_i(t)}$  quadrature amplitude modulation is given as  $\phi_i(t) \geq 4$ . The required transmit power of the ground node depends on  $\phi_i(t)$  and  $h_i(\zeta(t))$ , and is given by  $P_i(t) \approx \frac{\kappa_2^{-1} \ln \frac{\kappa_1}{\varepsilon}}{\|h_i(\zeta(t))\|^2} (2^{\phi_i(t)} - 1)$ , where  $\kappa_1$  and  $\kappa_2$  are channel constants, and  $\varepsilon$  defines the required bit error rate (BER) of the channel [16].

We consider that the UAV moves in low altitude for the data collection, where the probability of LoS communication between the UAV and the ground nodes can be

$$\Pr_{\text{LoS}}(\varphi_i) = \frac{1}{1 + a \exp(-b[\varphi_i - a])} \quad (1)$$

where  $a$  and  $b$  are two Sigmoid function parameters.  $\varphi_i$  is an elevation angle between the UAV and ground node  $i$ . Furthermore, the path loss of the link between the UAV and node  $i$  can be obtained by

$$\gamma_i = \Pr_{\text{LoS}}(\varphi_i)(\eta_{\text{LoS}} - \eta_{\text{NLoS}}) + 20 \log(r \sec \varphi_i) + 20 \log(\lambda) + 20 \log(4\pi/v_c) + \eta_{\text{NLoS}} \quad (2)$$

where  $r$  is the radius of the radio coverage of the UAV,  $\lambda$  is the carrier frequency, and  $v_c$  is the speed of light.  $\eta_{\text{LoS}}$  and  $\eta_{\text{NLoS}}$  represent the excessive path loss of LoS or non-LoS, e.g., the value of  $(\eta_{\text{LoS}}, \eta_{\text{NLoS}})$  pair can be (0.1, 21), (1.0, 20), (1.6, 23), and (2.3, 34) corresponding to suburban, urban, dense urban, and highrise urban scenarios, respectively [17].

Although multi-user beamforming techniques such as zero forcing and maximal ratio transmission can be used for air-ground communications to increase SNR of the channel, they are not considered in this work due to the requirement of real-time feedback on the channel conditions.

## III. MDP FOR FLIGHT RESOURCE ALLOCATION

The flight resource allocation problem is formulated as an MDP. The network states consist of battery levels  $e_i(t)$  and data queue lengths  $q_i(t)$  of each ground node, SNR of the channel  $h_i(\zeta(t))$ , and the location of UAV  $\zeta(t)$ . In particular, the flight altitude of the UAV is maintained. In other words, the UAV moves on a two-dimensional plane above the ground. To minimize the packet loss, the UAV takes actions to adjust the heading  $\psi_{\zeta(t)}$  and the patrol velocity  $v_{\zeta(t)}$  while selecting the ground nodes for data collection. At each waypoint  $\zeta(t)$ ,  $\psi_{\zeta(t)}$  takes real numbers in a continuous action space, i.e.,  $\psi_{\zeta(t)} \in (0, 2\pi]$ . Similarly, we have  $v_{\zeta(t)} \in (0, V]$ , where  $V$  is the highest patrol velocity of the UAV.

The future energy level and queue length of every ground node can be affected by the flight resource allocation, which also leads to a non-negligible impact on the future actions of the UAV. The actions of the UAV can be optimized in a long-term stochastic control process, where the optimality is achieved in regards of a specific metric, e.g., packet loss stemming from both overflowing buffers and unsuccessful data transmissions of the ground nodes [18]. Moreover,

the actions of the UAV are synthetically determined by the random data arrival or queueing status at the ground node, the heading and velocity control, and node selection decisions taken by the UAV. The correlation between the actions of the UAV in different time slots needs to be captured, and validate the long-term optimality.

An MDP is defined by the quadruplet  $\langle \mathcal{S}, \mathcal{A}, C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}, \Pr\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\} \rangle$ , where  $\mathcal{S}$  is the network state space, collecting  $e_i(t)$ ,  $q_i(t)$ ,  $\zeta(t)$ , and  $\mathbf{h}_i(\zeta(t))$ ; and  $\mathcal{A}$  is the set of actions to be taken by the UAV, i.e.,

$$\mathcal{A} = \left\{ \psi_{\zeta(t)}, v_{\zeta(t)}, (i, \phi_i(t)) : i = 1, 2, \dots, N; \right. \\ \left. \phi_i(t) \in \{1, 2, \dots, \Phi\} \right\}. \quad (3)$$

Let  $\mathcal{S}_\beta$  denote the following network state of  $\mathcal{S}_\alpha$  when action  $k \in \mathcal{A}$  is taken by the UAV. The immediate cost from  $\mathcal{S}_\alpha$  to  $\mathcal{S}_\beta$  can be given by  $C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}$ , while the transition probability is  $\Pr\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}$ .

The optimal policies in MDP can be computed by value iteration (iteratively improves an estimated action-value function) or policy iteration (explores a new policy iteratively and updates the action-value function under this new policy). However,  $C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}$  and  $\Pr\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}$  have to be known a-priori for obtaining the action-value function in value iteration and policy iteration. In contrast, this letter focuses on a practical scenario where the prior information of  $C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}$  and  $\Pr\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k\}$  is unknown.

#### IV. ONBOARD DDPG-FRA FRAMEWORK

An action-value function can be defined to minimize the expected network cost when the UAV takes an action following one of the resource allocation strategies thereafter. Specifically, the action-value function of each resource allocation strategy can be expressed in the form of the expected packet loss of the ground nodes at the current state  $\mathcal{S}_\alpha$  and the minimum of discounted  $Q\{\mathcal{S}_\beta, k'\}$  over all future network states, i.e.,

$$Q^*\{\mathcal{S}_\alpha, k^*\} = (1 - \varrho)Q^*\{\mathcal{S}_\alpha, k^*\} + \\ \varrho \left[ C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k^*\} + \delta \min_{k' \in \mathcal{A}} Q\{\mathcal{S}_\beta, k'\} \right] \quad (4)$$

where  $k'$  is the following action of  $k$ ,  $\delta \in [0, 1]$  is a discount factor, and  $\varrho \in (0, 1]$  is the learning rate.

It is known that Q-learning suffers from the curse-of-dimensionality, where the state and action spaces have to be discrete and low dimension, and cannot be applied to large continuous action spaces, e.g., continuous heading and patrol velocity control of the UAV. To address this issue, DDPG-FRA enables the UAV to construct four neural networks based on an actor-critic framework [19], i.e.,  $Q^A\{\mathcal{S}_\alpha, k | \theta^A\}$  parameterized by weights  $\theta^A$ , a target Q-network  $Q^{A'}\{\mathcal{S}_\alpha, k | \theta^{A'}\}$  parameterized by weights  $\theta^{A'}$ , a deterministic policy network  $Q^B\{\mathcal{S}_\alpha | \theta^B\}$ , and a target deterministic policy network  $Q^{B'}\{\mathcal{S}_\alpha | \theta^{B'}\}$  which

#### Algorithm 1 Onboard DDPG-FRA Framework

- 1: **1. Initialize:**
- 2:  $\mathcal{S}_\alpha \in \mathcal{S}$ ,  $k \in \mathcal{A}$  in (3),  $Q^A\{\mathcal{S}_\alpha, k | \theta^A\}$ ,  $Q^B\{\mathcal{S}_\alpha | \theta^B\}$ ,  $Q^{A'}\{\mathcal{S}_\alpha, k | \theta^{A'}\}$ ,  $Q^{B'}\{\mathcal{S}_\alpha | \theta^{B'}\}$ , where  $\theta^A \rightarrow \theta^{A'}$  and  $\theta^B \rightarrow \theta^{B'}$ .
- 3: Learning time  $\rightarrow t_{\text{learning}}$ . Experience replay capacity  $\rightarrow C_{\text{replay}}$ .
- 4: **2. Learning:**
- 5: **for** episode 1  $\rightarrow J$  **do**
- 6: The UAV observes  $\mathcal{S}_\alpha$ . Action exploration noise  $\rightarrow \Gamma_{\text{UAV}}$ .
- 7: **while**  $t \leq t_{\text{learning}}$  **do**
- 8: The UAV carries out action  $k_t \in \mathcal{A}$ , where  $k_t = Q^B\{\mathcal{S}_\alpha | \theta^B\} + \Gamma_{\text{UAV}}$ , which sets  $\psi_{\zeta(t)}$  and  $v_{\zeta(t)}$  of the UAV, and selects a ground node.
- 9: The UAV  $\leftarrow C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k_t\}$ , and obtains a new observation  $\mathcal{S}_\beta$ .
- 10: At the UAV:  $(\mathcal{S}_\alpha, \mathcal{S}_\beta, k_t, C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k_t\}) \rightarrow C_{\text{replay}}$ .
- 11: The UAV randomly takes a minibatch of  $F$  samples from the onboard memory  $C_{\text{replay}}$ .
- 12: For each sample  $f$ , we have  $y_f = (C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k_t\}, \mathcal{S}_\beta)_f + \delta Q^{A'}\{\mathcal{S}_{f+1}, Q^{B'}\{\mathcal{S}_{f+1} | \theta^{B'}\} | \theta^{A'}\}$ .
- 13: Minimizing a loss function onboard UAV, where  $\Delta_{\text{loss}} \leftarrow \frac{1}{F} \sum_f (y_f - Q^A\{\mathcal{S}_f, k_f | \theta^A\})^2$ .
- 14: Computing policy update onboard at the UAV, where  $\nabla_{\theta^B} \approx \frac{1}{F} \sum_f \nabla_k Q^A\{\mathcal{S}_\alpha, k_f | \theta^A\} \nabla_{\theta^B} Q^B\{\mathcal{S}_\alpha | \theta^B\} \big|_{\mathcal{S}_\alpha = \mathcal{S}_f}$ .
- 15: The UAV updates the two onboard target neural networks, where  $\theta^{A'} \leftarrow \epsilon \theta^A + (1 - \epsilon) \theta^{A'}$  and  $\theta^{B'} \leftarrow \epsilon \theta^B + (1 - \epsilon) \theta^{B'}$ .
- 16: **end while**
- 17: **end for**

specifies the current policy by deterministically mapping states to a specific action. The use of  $Q^{A'}\{\mathcal{S}_\alpha, k | \theta^{A'}\}$  and  $Q^{B'}\{\mathcal{S}_\alpha | \theta^{B'}\}$  in DDPG-FRA can reduce the approximation errors and regularize the actions of the UAV, which increases learning stability.

Algorithm 1 presents the implementation of DDPG-FRA, which optimizes the flight resource allocation onboard at the UAV for minimizing the overall data packet loss of the ground nodes. Given a total of  $J$  episodes and a learning time of  $t_{\text{learning}}$ , the UAV explores the next flight resource allocation action according to a policy  $k_t = Q^B\{\mathcal{S}_\alpha | \theta^B\} + \Gamma_{\text{UAV}}$ , where  $\Gamma_{\text{UAV}}$  is an action exploration noise in the environment. By carrying out the action  $k_t$ , the UAV can obtain the next state  $\mathcal{S}_\beta$  as well as  $C\{\mathcal{S}_\beta | \mathcal{S}_\alpha, k_t\}$ . Moreover, the flight resource allocation experience sampled from the environment can be stored in a memory with the capacity of  $C_{\text{replay}}$  onboard at the

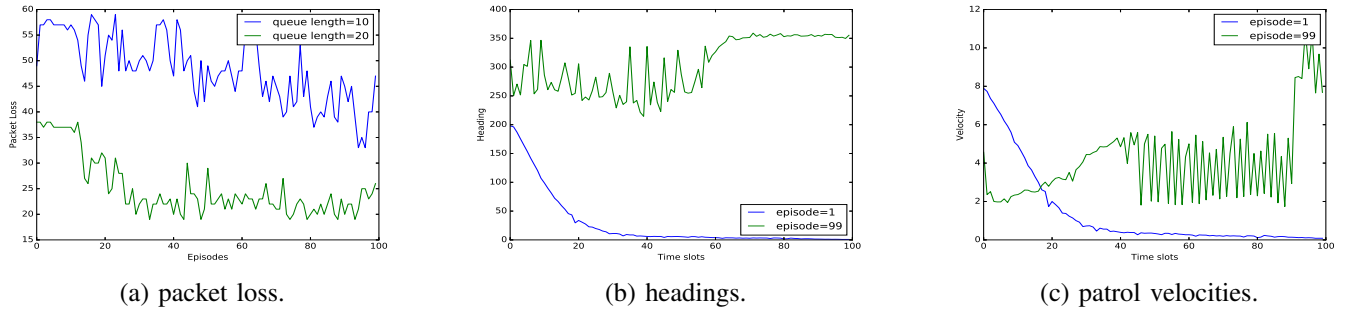


Fig. 2: (a) Network packet loss at the episode given data buffer size of 10 or 20 packets. (b) At the episode 1 or 99, headings of the UAV in regards to each time slot. (c) At the episode 1 or 99, patrol velocities of the UAV in regards to each time slot.

UAV for the experience replay. A large  $C_{replay}$  allows the DDPG-FRA framework to benefit from learning across a set of uncorrelated experiences.

To train the onboard neural networks, DDPG-FRA randomly samples a minibatch of  $(\mathcal{S}_\alpha, \mathcal{S}_\beta, k_t, C\{\mathcal{S}_\beta|\mathcal{S}_\alpha, k_t\})$  from the experience replay memory. The UAV optimizes the values of  $\theta^A$  to minimize the gap between  $Q^A$  and  $Q^{A'}$ , which is denoted by  $\Delta_{loss}$ . Therefore, the optimal actions of the UAV can be specified according to (4). Furthermore, a policy can be generated and trained with the updated  $\theta^A$  onboard at the UAV, where

$$\nabla_{\theta^B} \approx \frac{1}{F} \sum_f \nabla_{k_f} Q^A \left\{ \mathcal{S}_\alpha, k_f \middle| \theta^A \right\} \nabla_{\theta^B} Q^B \left\{ \mathcal{S}_\alpha \middle| \theta^B \right\} \Big|_{\mathcal{S}_\alpha = \mathcal{S}_f} \quad (5)$$

Let  $\epsilon \in (0, 1)$  denote a learning factor that responds the update of the target networks. Given the updated policy  $\nabla_{\theta^B}$ , the UAV can accordingly update the two target neural networks by

$$\theta^{A'} = \epsilon \theta^A + (1 - \epsilon) \theta^{A'}, \quad (6)$$

$$\theta^{B'} = \epsilon \theta^B + (1 - \epsilon) \theta^{B'}. \quad (7)$$

By learning  $\nabla_{\theta^B}$  and minimizing  $\Delta_{loss}$ ,  $\psi_{\zeta(t)}$  and  $v_{\zeta(t)}$  of the UAV are optimized at  $\mathcal{S}_\alpha$ , while the optimal ground node is scheduled for data transmission. Given the optimal actions, the next state  $\mathcal{S}_\beta$  that is determined by the battery levels and queue lengths of all other unselected ground nodes can be accordingly updated. Iteratively, the proposed DDPG-FRAS explores all the network states during  $t_{learning}$ , and determines the optimal action of the UAV at each state.

## V. NUMERICAL RESULTS

DDPG-FRAS is implemented in Python 3.5 on PyTorch which is an open source machine learning library based on the Torch library [20]. A laptop with 8GB RAM and an Intel Core i5-7200U based on 64-bit Ubuntu 16.04 is used for the PyTorch setup. The area of interest is set to be a square area with a size of 1,000 m  $\times$  1,000 m, and  $N$  ground nodes are distributed in the region, where  $N = 40$ . Each ground node has the maximum discretized battery capacity of 50 Joules, the highest modulation of  $\Phi$

$= 5$ , and the maximum transmit power of 100 milliwatts. For calculating  $P_i(t)$  of the ground node, the two channel constants,  $\kappa_1$  and  $\kappa_2$  are set to 0.2 and 3, respectively. The required BER is  $\epsilon = 0.05\%$ , and the carrier frequency  $\lambda$  is 2000 MHz. The UAV has the highest patrol velocity  $V = 15$  m/s. The heading  $\psi_{\zeta(t)}$  defines the angle with the north direction, which can be adjusted between  $0^\circ$  and  $360^\circ$ .

Figure 2(a) shows the network packet loss at each training episode of the proposed DDPG-FRAS, given the buffer length of 10 or 20 packets per device. Generally, DDPG-FRAS with  $L = 20$  achieves a lower packet loss than the one with  $L = 10$  for around 25 packets. This is because DDPG-FRAS significantly reduces buffer overflow for all the ground nodes when enlarging their data buffer size. Moreover, DDPG-FRAS has a large network packet loss at the beginning of training the onboard DDPG. With an increasing number of episodes, the network packet loss drops significantly until it reaches a relatively stable value. It confirms that the onboard DDPG can gradually converge after the flight resource allocation is sufficiently trained. Figures 2(b) and 2(c) plot the headings and patrol velocities of the UAV at the first and the 99th episode, respectively, where each episode consists of 100 training time slots. As observed, the flight of the UAV is adapted to the training process of DDPG-FRAS. In particular, the flight is hardly optimized at the first episode since the experience in the replay memory is not sufficient for a small number of learning iterations. At the 99th episode,  $Q^A\{\mathcal{S}_\alpha, k|\theta^A\}$ ,  $Q^{A'}\{\mathcal{S}_\alpha, k|\theta^{A'}\}$ ,  $Q^B\{\mathcal{S}_\alpha|\theta^B\}$ , and  $Q^{B'}\{\mathcal{S}_\alpha|\theta^{B'}\}$  are adequately trained by Algorithm 1 to minimize the loss functions by taking advantage of the experience replay, as can also be observed in Figure 2(a).

Figure 3 studies the packet loss achieved by the proposed DDPG-FRAS scheme with an increasing number of ground nodes. In particular, the buffer size of the ground node is set to 10, 15, or 20 given the average SNR of 0 dB or 18 dB. Generally, the packet loss grows with the network size since more ground nodes have to buffer their data while one node is selected by the UAV to transmit data. DDPG-FRAS slightly improves the performance at a higher SNR. For example, for 20 ground nodes, the packet loss



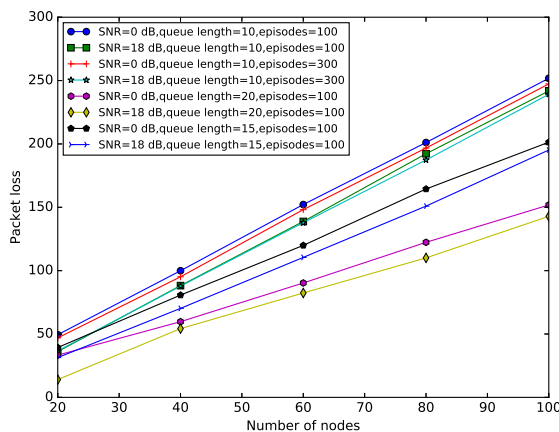


Fig. 3: Packet loss achieved by DDPG-FRAS with regards to number of ground nodes.

of (SNR = 18 dB, queue length = 20, episodes = 100) is lower than (SNR = 0 dB, queue length = 20, episodes = 100) for 12 packets. Extending the learning episodes barely reduces the packet loss. This confirms that the packet loss in the online flight resource allocation maintains stable once DDPG-FRAS converges, as observed in Figure 2(a).

Given the same number of learning episodes and average SNR, enlarging the buffer size can significantly reduce the packet loss for DDPG-FRAS. For example, when the number of ground nodes is 100, SNR = 0 dB and episodes = 100, the packet loss of DDPG-FRAS with queue length = 20 is lower than the one with queue length = 15 and queue length = 10 for 50 and 115 packets, respectively. This implies that the buffer size of the ground node has a dominating effect on DDPG-FRAS.

## VI. CONCLUSION

This letter studies an online flight resource allocation problem for jointly controlling the UAV's flight and scheduling data transmissions of the ground sensor nodes, to prevent buffer overflow and unsuccessful data transmissions. The flight resource allocation problem can be formulated as an MDP. We propose the DDPG-FRA strategy to optimize the flight resource allocation, where the up-to-date knowledge of battery levels, data queue lengths, and channel conditions of the ground nodes is not available at the UAV. Continuous action spaces of headings and patrol velocities of the UAV are implemented. Numerical results demonstrate that headings and velocities of the UAV can be efficiently trained by DDPG-FRA to minimize the data packet loss of the ground nodes, by taking advantage of onboard experience replay.

## ACKNOWLEDGEMENTS

This work was partially supported by National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology), within the CISTER Research Unit (UIDB/04234/2020); also by the Operational Competitiveness Programme and Internationalization (COMPETE

2020) under the PT2020 Partnership Agreement, through the European Regional Development Fund (ERDF), and by national funds through the FCT, within project(s) POCI-01-0145-FEDER-029074 (ARNET).

## REFERENCES

- [1] V. Chamola, V. Hassija, V. Gupta, and M. Guizani, "A comprehensive review of the COVID-19 pandemic and the role of IoT, drones, ai, blockchain, and 5G in managing its impact," *IEEE Access*, vol. 8, pp. 90 225–90 265, 2020.
- [2] L. Li, Y. Xu, Z. Zhang, J. Yin, W. Chen, and Z. Han, "A prediction-based charging policy and interference mitigation approach in the wireless powered internet of things," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 2, pp. 439–451, Feb. 2019.
- [3] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, and W. Zhuang, "UAV relay in VANETs against smart jamming with reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4087–4097, May 2018.
- [4] S. Corp., "Softbank corp. develops aircraft that delivers telecommunications connectivity from the stratosphere." [Online]. Available: [https://www.softbank.jp/en/corp/news/press/sbkk/2019/20190425\\_02/](https://www.softbank.jp/en/corp/news/press/sbkk/2019/20190425_02/)
- [5] Z. Liu, "Chinese solar-powered drone morning star spreads its wings in successful test flight." [Online]. Available: <https://www.scmp.com/news/china/military/article/2171081/chinese-solar-powered-drone-spreads-its-wings-successful-test>
- [6] K. Li, W. Ni, L. Duan, M. Abolhasan, and J. Niu, "SWPT: A joint-scheduling model for wireless powered sensor networks," in *IEEE Global Communications Conference (GLOBECOM)*, 2017.
- [7] X. Yu, X. Zhou, and Y. Zhang, "Collision-free trajectory generation for UAVs using markov decision process," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017.
- [8] S. Ragi and E. K. Chong, "UAV guidance algorithms via partially observable markov decision processes," *Handbook of Unmanned Aerial Vehicles*, pp. 1775–1810, 2015.
- [9] D. Cao, Z. Yin, W. Yang, and G. Kang, "An energy-efficient transmission scheme for buffer-aided UAV relaying networks," in *IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*. IEEE, 2019, pp. 1–5.
- [10] K. Li, W. Ni, L. Duan, M. Abolhasan, and J. Niu, "Wireless power transfer and data collection in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2686–2697, Mar. 2018.
- [11] K. Li, W. Ni, M. Abolhasan, and E. Tovar, "Reinforcement learning for scheduling wireless powered sensor communications," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 2, pp. 264–274, Jun. 2018.
- [12] K. Li, W. Ni, B. Wei, and E. Tovar, "Onboard double Q-learning for airborne data capture in wireless powered IoT networks," *IEEE Networking Letters*, vol. 2, no. 2, pp. 71–75, 2020.
- [13] K. Li, W. Ni, E. Tovar, and A. Jamalipour, "On-board deep Q-network for UAV-assisted online power transfer and data collection," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 12 215 – 12 226, Oct. 2019.
- [14] Z. M. Fadlullah, D. Takaishi, H. Nishiyama, N. Kato, and R. Miura, "A dynamic trajectory control algorithm for improving the communication throughput and delay in UAV-aided networks," *IEEE Netw.*, vol. 30, no. 1, pp. 100–105, Jan. 2016.
- [15] Z. Han, A. L. Swindlehurst, and K. R. Liu, "Smart deployment/movement of unmanned air vehicle to improve connectivity in MANET," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2006.
- [16] K. Li, W. Ni, X. Wang, R. P. Liu, S. S. Kanhere, and S. Jha, "Energy-efficient cooperative relaying for unmanned aerial vehicles," *IEEE Trans. Mobile Comput.*, vol. 15, no. 6, pp. 1377–1386, Jun. 2016.
- [17] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, Dec. 2014.
- [18] Y. Emami, K. Li, and E. Tovar, "Buffer-aware scheduling for UAV relay networks with energy fairness," in *IEEE Vehicular Technology Conference (VTC Spring)*, 2020.
- [19] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *International Conference on Machine Learning (ICML)*, 2014.
- [20] N. Ketkar, "Introduction to PyTorch," in *Deep learning with Python*. Springer, 2017, pp. 195–208.