

---

This is an electronic reprint of the original article.  
This reprint may differ from the original in pagination and typographic detail.

Hellaoui, Hamed; Yang, Bin; Taleb, Tarik

## Towards using Deep Reinforcement Learning for Connection Steering in Cellular UAVs

*Published in:*

2021 IEEE Global Communications Conference, GLOBECOM 2021 - Proceedings

*DOI:*

[10.1109/GLOBECOM46510.2021.9685265](https://doi.org/10.1109/GLOBECOM46510.2021.9685265)

Published: 01/01/2021

*Document Version*

Peer reviewed version

*Please cite the original version:*

Hellaoui, H., Yang, B., & Taleb, T. (2021). Towards using Deep Reinforcement Learning for Connection Steering in Cellular UAVs. In *2021 IEEE Global Communications Conference, GLOBECOM 2021 - Proceedings (2021 IEEE Global Communications Conference, GLOBECOM 2021 - Proceedings)*. IEEE.  
<https://doi.org/10.1109/GLOBECOM46510.2021.9685265>

---

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

© 2022 IEEE. This is the author's version of an article that has been published by IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

# Towards using Deep Reinforcement Learning for Connection Steering in Cellular UAVs

Hamed Hellaoui<sup>1</sup>, Bin Yang<sup>1</sup>, and Tarik Taleb<sup>1,2</sup>

<sup>1</sup>Communications and Networking Department, Aalto University, Finland.

<sup>2</sup>University of Oulu, 90570 Oulu, Finland.

Email: {hamed.hellaoui, bin.l.yang, tarik.taleb}@aalto.fi

**Abstract**—This paper investigates the fundamental connection steering issue in cellular-enabled Unmanned Aerial Vehicles (UAVs), whereby a UAV steers the cellular connection across multiple Mobile Network Operators (MNOs) for ensuring enhanced Quality-of-Service (QoS). We first formulate the issue as an optimization problem for minimizing the maximum outage probability. This is a nonlinear and nonconvex problem that is generally difficult to be solved. To this end, we propose a new approach for solving the optimization problem based on Deep Reinforcement Learning (DRL), considering two important reinforcement learning algorithms (i.e., Deep Q-Learning (DQN) and Advantage Actor Critic (A2C)). Simulation results show that under the proposed approach, the UAVs can make optimal decisions to select the most suitable connection with MNOs for achieving the minimization of the maximum outage probability. Furthermore, the results also show that in our new approach, the A2C-based algorithm is better than the DQN-based one, especially when the number of MNOs increases, while the DQN-based algorithm can be executed in a shorter time.

**Index Terms**—Unmanned Aerial Vehicles (UAVs), 5G, Beyond 5G, Mobile Networks, Connection Steering, and Deep Reinforcement Learning.

## I. INTRODUCTION

Mobile networks have been advocated to be the communication infrastructure to support the challenging applications of Unmanned Aerial Vehicles (UAVs) [1]. Specifically, UAVs, which connect to cellular networks, have attracted increasing attention from both military and civilian fields like remote monitor, industrial detection, cargo delivery and remote sensing. This is because cellular UAVs can provide distant communication services with high throughput, low delay and strong security, and thus can satisfy various application requirements. Accordingly, they have been envisioned as a critical component in the 5<sup>th</sup> generation of mobile networks and beyond.

The cellular UAVs could bring many new opportunities. Particularly, the UAVs can flexibly switch their connections with different network operators (MNOs) within their coverage range for routing the traffic, aiming to significantly improve the Quality-of-Services (QoS). Indeed, some new OBUs (On-Board Units), which are used to enable vehicular communications to cellular networks, integrate the possibility to connect to several mobile networks at the same time. As illustrated in Fig. 1, an OBU can support the simultaneous connections to several mobile networks [2]. The traffic from the OBU is first directed, using one selected MNO, to a steering

service residing in an edge service nearby the base stations (BSs) of the concerned MNOs. This service ensures seamless connection to the internet by preserving one IP address if a steering operation happens. Although an OBU module is being originally considered for vehicular communications, this principle can be also considered for UAVs. A crucial issue for these OBU-equipped mobile UAVs is how to constantly select and connect to the right MNO for ensuring enhanced QoS and accordingly steering the traffic.

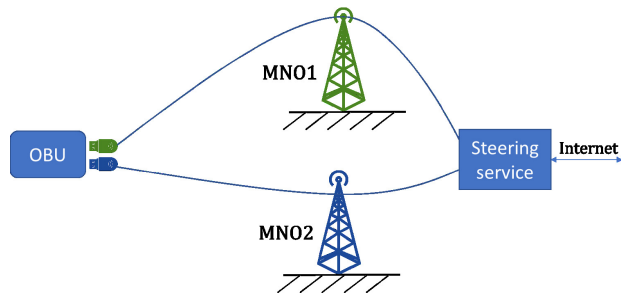


Fig. 1: An OBU module can enable the connection to several mobile networks at the same time.

While the concept of traffic steering among multiple MNOs can enhance the QoS for cellular UAVs, it comes with important challenges. Effectively, the selection of MNOs is of fundamental importance to achieve enhanced QoS. This selection depends on different parameters and becomes more complex in case of a large scale network. Furthermore, given the moving nature of UAVs, the traffic steering decision needs to be taken within a relatively short time. This underpins the focus of this paper, where we elaborate on enabling traffic steering for cellular UAVs in online use. To this end, we advocate an approach based on Deep Reinforcement Learning (DRL). Recently, available works on DRL for cellular UAVs mainly focus on the studies of path planning and resource management. However, to the best of the authors' knowledge, no work has considered DRL for traffic steering, particularly in the context of cellular UAVs.

The rest of this paper is organized in the following fashion. We review some related works on connection steering and reinforcement learning in Section II. The considered system model and the formulation of the problem are presented in Section III. Thereafter, we introduce the proposed deep rein-

forcement learning approach for traffic steering in Section IV. Performance evaluations are provided in Section V. The paper concludes in Section VI.

## II. RELATED WORK

In the literature, some works considered the concept of connection steering to route the traffic between different network functions [3], [4]. However, this principle is less studied for the part between the connected devices and the serving BSs. The work in [5] focused on LTE-connected vehicles. The authors target enhancing the communication by anticipating QoS degradation and directing the traffic to different Radio Access Technologies (RAT). In [6], the authors considered the problem of connection steering in cellular-enabled UAVs. The proposed solution steers UAV communication to the mobile network ensuring the best Radio Signal Strength Indicator (RSSI) quality. The work is analyzed by applying Discrete Time Markov Chain (DTMC) to evaluate the performance of the testbed results. However, while RSSI can be considered as a good indicator for terrestrial communication, aerial communication presents different characteristics imposing the revision of such indicators. In [7], the authors proposed a coalitional game-based solution for traffic steering in cellular UAVs. The goal is to form UAVs in coalitions around MNOs in a way to enhance their QoS. However, the convergence of the game takes time which makes such a solution more adequate for offline use (planning) rather than for online use.

Recently some studies have proposed the use of DRL for cellular UAVs, mainly for path planning and resource management. In [8], the authors proposed a DRL for cellular UAVs. The goal is to optimize UAV path planning while taking resource management into consideration by achieving a trade-off between maximizing energy efficiency and minimizing both wireless latency and the interference. The authors in [9] tackled the problem of UAV navigation in a way to have the best UAV-ground link. The authors considered massive multiple-input-multiple output (MIMO) and proposed a deep Q-learning for selecting the optimal policy. In [10], the authors considered the application of providing wireless charging for UAVs deployed to collect data from sensor devices scattered in the physical environment. A reinforcement learning based approach is proposed to plan the route of a UAV, where the problem is formulated as a Markov decision process and Q-learning is used to find the optimal policy. The authors in [11] considered the problem of providing computation resources to ground UE using Flying Mobile Edge Computing (F-MEC) on top of UAVs. A reinforcement learning based algorithm is proposed to optimize the trajectory of the UAVs. In another work [12], the authors focused on network aided UAVs, where UAVs serve as aerial base stations for multiple ground users. The trajectory design is investigated to maximize the expected uplink sum rate with inaccessibility to user-side information, such as locations and transmit power as well as channel parameters. The problem is formulated as a Markov decision process and is solved with model-free reinforcement learning.

Although the application of DRL for cellular UAVs is getting more interest, their use for traffic steering has not been investigated, and that is to the best of the authors' knowledge. Such an application is very crucial, mainly to enable quick and online decisions for flying UAVs. In the next section, we present the system model for traffic steering in cellular UAVs as well as the problem formulation.

## III. SYSTEM MODEL AND PROBLEM FORMULATION

1) *System Model:* We consider a cellular network consisting of UAVs and BSs. BSs belong to different MNOs. We use  $\mathcal{O}$ ,  $\mathcal{U}$  and  $\mathcal{V}_o$  to denote the sets of MNOs, UAVs and BSs, respectively. We also denote by  $\mathcal{C}_o$  the set of sub-carrier used by the MNO  $o \in \mathcal{O}$ . As shown in Fig 2, each UAV  $u \in \mathcal{U}$  has a serving BS in each MNO, and can steer its communication to only one selected MNO. We also consider the uplink scenario and we use the term  $uv_o$  to denote the link between the UAV  $u \in \mathcal{U}$  and the BS  $v_o \in \mathcal{V}_o$ , while the term  $tv_o$  is used to denote the link between the interfering UAV  $t \in \mathcal{U}$  and the non serving BS  $v_o \in \mathcal{V}_o$ . The instantaneous received signal-to-noise ratio (SNR) for the link  $uv_o$ , which is denoted by  $\gamma_{uv_o}$ , can be computed as

$$\gamma_{uv_o} = P_u \alpha_{uv_o}^2 / N_0, \quad (1)$$

where  $P_u$  stands for the transmission power of UAV  $u$ ,  $\alpha_{uv_o}$  is the fading coefficient and  $N_0$  refers to the variance of a zero-mean additive white Gaussian process. The instantaneous received signal-to-interference-plus-noise ratio (SINR) for the link between a UAV  $u$  and the BS  $v_o$  can be obtained as

$$SINR_{uv_o} = \gamma_{uv_o} / (1 + \sum_{\substack{t \neq u \\ t \in \mathcal{U}}} \gamma_{tv_o}). \quad (2)$$

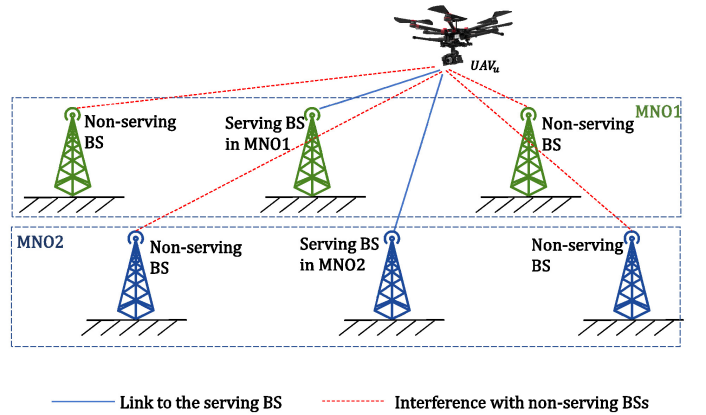


Fig. 2: System model (uplink): a UAV can connect to different MNOs and steer the connection only via one selected MNO.

We consider a probabilistic model for the propagation channel which depends on the line-of-sight (LoS) condition  $P_{uv_o}^{LoS}$  defined in 3GPP [13]. The LoS situation results in better channel conditions for the UAV. The path loss expression,  $PL_{uv_o}$ , is therefore expressed in terms of this condition introduced in [13]. We also take into account the effect of

fading. It follows a Nakagami- $m$  distribution for LoS links, and a Rayleigh distribution for NLoS links. We define the two parameters  $A_{uv_o}$  and  $B_{uv_o}$  to characterize the mean values of the SNR, for LoS and NLoS conditions, respectively, as

$$\begin{cases} A_{uv_o} &= P_{uv_o}^{LoS} \times P_u/N_0 \times 10^{-\frac{PL_{uv_o}}{10}}, \\ B_{uv_o} &= P_{uv_o}^{NLoS} \times P_u/N_0 \times 10^{-\frac{PL_{uv_o}}{10}}. \end{cases} \quad (3)$$

**Theorem 1.** *For an uplink communication from a UAV  $u$  to the BS  $v_o \in \mathcal{V}_o$ , we use outage probability  $P_{uv_o}^{out}(\gamma_{th})$  to characterize the probability that SINR $_{uv_o}$  falls below a target threshold  $\gamma_{th}$  leading to the failure of data transmission between  $u$  and  $v_o$ . Then, it can be expressed as*

$$\begin{aligned} P_{uv_o}^{out}(\gamma_{th}) &= \sum_{j=1}^m \left( \beta_{1j} \frac{(-1)^j}{(j-1)!} \left( \frac{m}{A_{uv_o}} \right)^{-j} \left( \Gamma(j) + \sum_{t=1}^N \delta'_t f_{j,1}(B_{tv_o}) - \right. \right. \\ &\quad \left. \left. \sum_{t=1}^N \sum_{j'=1}^m \delta_{t,j'} \frac{(-1)^{j'}}{(j'-1)!} f_{j,j'}(A_{tv_o}/m) \right) \right) - \beta_{21} B_{uv_o} \left[ 1 + \exp\left(-\frac{\gamma_{th}}{B_{uv_o}}\right) \right. \\ &\quad \left. \left( \sum_{t=1}^N \frac{\delta'_t}{\frac{\gamma_{th}}{B_{uv_o}} + \frac{1}{B_{tv_o}}} - \sum_{t=1}^N \sum_{j=1}^m \frac{\delta_{t,j}}{\left(\frac{\gamma_{th}}{B_{uv_o}} + \frac{m}{A_{tv_o}}\right)^j (j-1)!} \Gamma(j) \right) \right] \end{aligned} \quad (4)$$

where  $\Gamma(j)$  is the gamma function. ( $[1, \dots, N]$ ) refers to the list of interferer UAVs. The terms  $\beta_{1j}$ ,  $\beta_{21}$ ,  $\delta'_t$  and  $\delta_{t,j}$  have unique values satisfying the following formulas (fractional decomposition)

$$\left(1 - x \frac{A_{uv_o}}{m}\right)^{-m} (1 - x B_{uv_o})^{-1} = \sum_{j=1}^m \frac{\beta_{1j}}{\left(x - \frac{m}{A_{uv_o}}\right)^j} + \frac{\beta_{21}}{\left(x - \frac{1}{B_{uv_o}}\right)} \quad (5)$$

$$\begin{aligned} &\prod_{t=1}^N (1 - x B_{tv_o})^{-1} \left(1 - x \frac{A_{tv_o}}{m}\right)^{-m} \\ &= \sum_{t=1}^N \frac{\delta'_t}{x - \frac{1}{B_{tv_o}}} + \sum_{t=1}^N \sum_{j=1}^m \frac{\delta_{t,j}}{\left(x - \frac{m}{A_{tv_o}}\right)^j}. \end{aligned} \quad (6)$$

The function  $f_{j,j'}(y)$  is provided as

$$f_{j,j'}(y) = \sum_{p=1}^n y^{j'} (\theta_p)^{j'-1} \lambda_p \Gamma\left(j, \frac{m\gamma_{th}(\theta_p y + 1)}{A_{uv_o}}\right), \quad (7)$$

where  $\lambda_p$  and  $\theta_p$  denote the weight and the zero factors of the  $n$ -th order Laguerre polynomials, respectively [14].  $\Gamma(a, z)$  is the upper incomplete gamma function defined as  $\Gamma(a, z) = \int_z^\infty t^{a-1} e^{-t} dt$ .

**Proof:** The proof is provided in [15]. ■

As mentioned in [15], Theorem 1 considers most of the propagation phenomena that the wireless signal undergoes, which makes the system model realistic.

2) *Problem Formulation:* As mentioned in the above subsection, each UAV can be connected to different MNOs and needs to steer the connection via one mobile network. The objective of this paper is to minimize the maximum outage probability by optimizing the selection of MNO for each UAV, which can reduce the outage probabilities for the worse-case

links for ensuring the QoS. To characterize the selection, we define the Boolean variable  $x_{uo}$  as

$$x_{uo} = \begin{cases} 1, & \text{If the UAV } u \text{ chooses the MNO } o \in \mathcal{O}, \\ 0, & \text{Otherwise.} \end{cases} \quad (8)$$

The steering problem can therefore be formulated as,

$$\min\text{-max}_{u \in \mathcal{U}} \left( \sum_{o \in \mathcal{O}} x_{uo} P_{uv_o}^{out}(\gamma_{th}) \right) \quad (9)$$

s.t.

$$\sum_{o \in \mathcal{O}} x_{uo} = 1, \quad \forall u \in \mathcal{U}, \quad (10)$$

$$x_{uo} \in \{0, 1\}, \quad \forall u \in \mathcal{U}, \forall o \in \mathcal{O}. \quad (11)$$

The objective function (9), expressed in the above optimization problem, aims to reduce the outage probability for the UAVs. This is subject to constraint (10) to ensure that each UAV selects one and only one MNO, and constraint (11) to limit the value of the decision variable to  $\{0, 1\}$ .

However, this optimization is not linear and complex to resolve, especially for a large network. Solving such a problem takes time. In order to enable quick and online traffic steering decisions for cellular UAVs, we propose an approach based on deep reinforcement learning. The next section introduces the proposed approach.

#### IV. DEEP REINFORCEMENT LEARNING FOR CONNECTION STEERING IN CELLULAR UAVS

While solving the above optimization problem can be considered for offline environments, this is not adequate for online use. To this end, we advocate in this paper a deep reinforcement approach to enable connection steering in cellular UAVs. DRL can be trained to learn complex tasks and effectively takes decisions through the interaction with the environment based on trial and error processes. More precisely, a RL agent periodically interacts with an environment, observes the current state  $s^t$ , then executes an action  $a^t$ . Subsequently, the agent will observe a new state  $s^{t+1}$  and receives a corresponding reward  $r^t$ . We also design a replay memory to store history of experiences that will be used during the learning process. Unlike supervised and unsupervised machine learning algorithms, RL techniques do not require prior dataset. In what follows, we present the architecture of the proposed DRL framework for connection steering. Thereafter, we introduce the underlying learning process.

##### A. Architecture of the Proposed DRL Framework

The general architecture of the proposed DRL framework is depicted in Fig. 3. In particular, we define the state of the system, the action space and the system reward.

1) *System State:* The system state is defined in a way to capture the feature of the current deployment. To this end, we consider the mean SNR to the serving BS of the selected MNO for each UAV in defining the system state. Furthermore, it is very important to define the system state

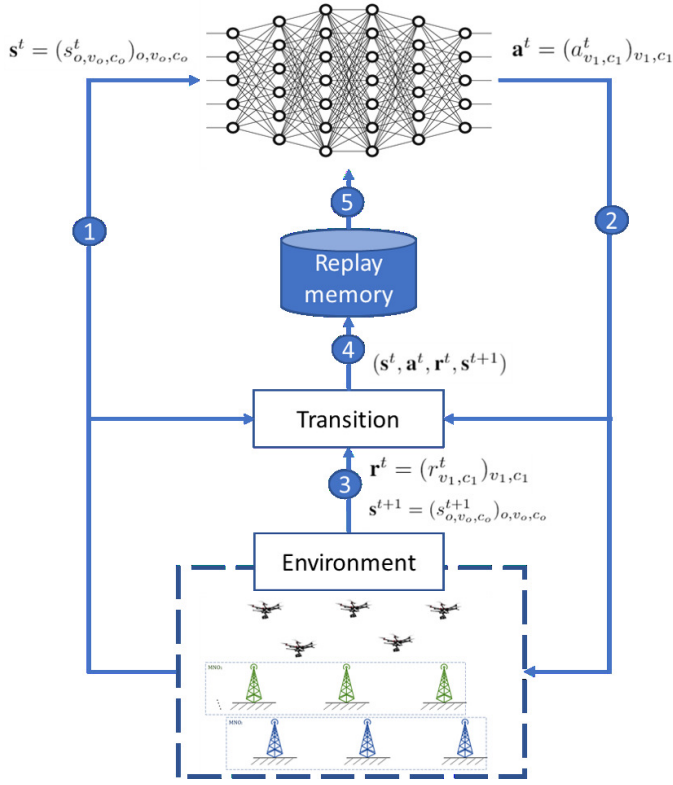


Fig. 3: Architecture of the DRL framework.

in a way to accommodate the dynamic of the network, so effective decisions can be made when the number of UAVs changes. In this regard, the number of MNOs, BSs, and sub-carriers does not change frequently in practice. Taking this into consideration, we define the function  $w_{(v_o, c_o)}$  that returns the UAV  $u$  being served by the MNO  $o$  and assigned with the sub-carrier  $c_o$  from the BS  $v_o \in \mathcal{V}_o$ . At a time step  $t$ , the system state can be defined as  $s^t = (s_{o, v_o, c_o}^t)_{o, v_o, c_o} \in \mathbb{R}^{|\mathcal{O}| \times |\mathcal{V}_o| \times |\mathcal{C}_o|}$ , where

$$s_{o, v_o, c_o}^t = \begin{cases} A_{uv_o} + B_{uv_o}, & \text{If } \exists u \in \mathcal{U} | u = w_{(v_o, c_o)}, \\ 0, & \text{Otherwise.} \end{cases} \quad (12)$$

As it can be seen, a system state is based on the mean SNR from each UAV to the serving BS of the selected MNO.

2) *Action Space*: After receiving a state, optimal selection of the target MNOs needs to be performed. The action therefore consists of the target MNOs to be selected by each UAV and is defined as  $a^t = (a_{v_1, c_1}^t)_{v_1, c_1} \in \mathcal{O}^{|\mathcal{V}_1| \times |\mathcal{C}_1|}$ . The actions are also applied to the UAVs in their assignment order to the first MNO. This allows to make a mapping between the captured state and the taken action, in terms of the order of the UAVs. This order will also be considered for the system reward as detailed in what follows.

3) *System Reward*: The goal is to select for each UAV the best MNO ensuring the enhanced QoS in the network. The system reward is therefore defined by considering the outage probabilities achieved by the UAVs after executing

the received actions. More precisely, a reward for a UAV  $u$  is based on  $P_{uv_o}^{out}(\gamma_{th})$ , where  $v_o$  is the served BS of the selected MNO after executing the action. The system reward is therefore defined as  $r^t = (r_{v_1, c_1}^t)_{v_1, c_1} \in [0, 1]^{|\mathcal{V}_1| \times |\mathcal{C}_1|}$ , where

$$r_{v_1, c_1}^t = \begin{cases} 1 - P_{uv_o}^{out}(\gamma_{th}), & \text{If } \exists u \in \mathcal{U} | u = w_{(v_1, c_1)}, \\ 1, & \text{Otherwise.} \end{cases} \quad (13)$$

As we can see, the system reward is based on the outage probabilities of UAVs according to their assignment order to the first MNO, as it is the case for the action.

### B. Learning Process

In the proposed framework, we consider two important reinforcement learning algorithms, namely Deep Q-Network (DQN) and Advantage Actor-Critic (A2C). A DQN agent relies on replaying experiences to ensure a stable learning. It uses Q-values (which is the maximum expected reward) and computes the temporal difference error based on the distance between Q-targets (which is the maximum value that can be captured from the next states) and the predicted Q-values. Two networks are used in our implementation of the DQN agent, namely Q-network and target network, to reduce the relevance between choosing actions and training the model [16]. In A2C algorithm, we consider two networks, i.e., the actor and critic networks. The Actor observes the environment and selects a given action by outputting a probability distribution across the action space. After that, the Critic evaluates the quality of the selected action regarding both the current state and the next state [17]. Furthermore, we use a replay memory to store experience in our framework, which is implemented as part of the agent.

Algorithm 1 summarizes the learning process adopted in the proposed approach. This process is executed in episodes until reaching a maximum value  $E$ . This process is common for both DQN and A2C agents. At each episode  $t$ , the agent gets the system state  $s^t$  from the environment. Thereafter, the agent selects an action  $a^t$ . This action is chosen based on the Q-network in the case of DQN agent, and based on the Actor network in the case of A2C agent. After executing the selected action, the agent gets the immediate reward  $r^t$  and the new state  $s^{t+1}$ . This allows to construct the transition  $(s^t, a^t, r^t, s^{t+1})$  and store it in the replay memory. Finally, the agent takes samples from the replay memory and learns from

---

#### Algorithm 1 DRL algorithm.

---

**Input:** Agent (DQN or A2C)

- 1: **for** episode  $t = 1$  **to**  $E$  **do**
  - 2:   Observe the state  $s^t$
  - 3:    $a^t = \text{Agent.select\_action}(s^t)$
  - 4:   Execute action  $a^t$
  - 5:   Get the reward  $r^t$
  - 6:   Observe the state  $s^{t+1}$
  - 7:    $\text{Agent.push\_replay\_memory}(s^t, a^t, r^t, s^{t+1})$
  - 8:    $\text{Agent.learn}()$
  - 9: **end for**
-

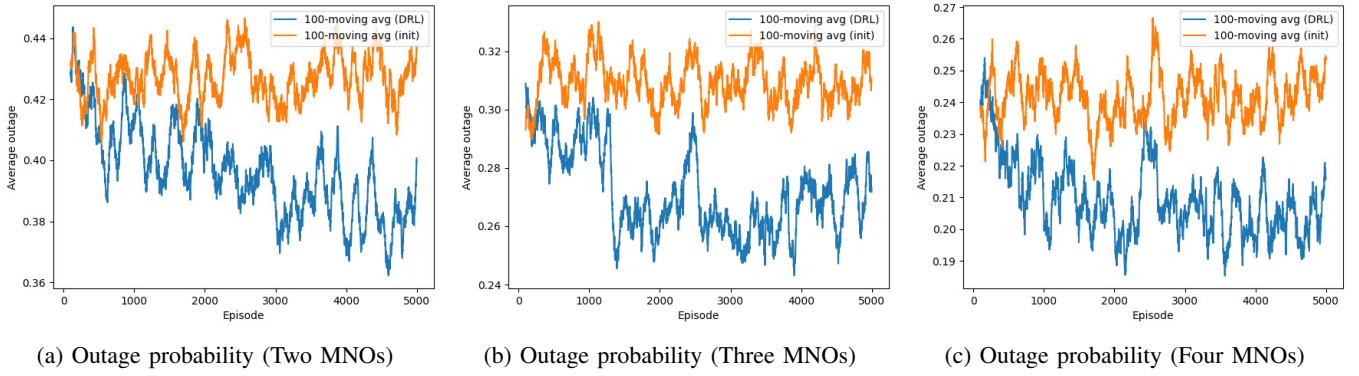


Fig. 4: Evaluation of the average reward and outage probabilities for DQN agent.

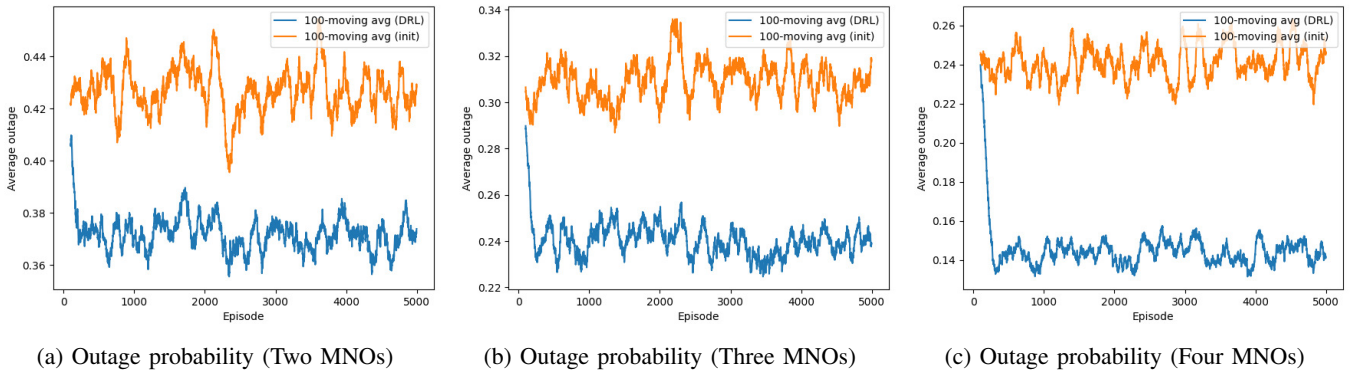


Fig. 5: Evaluation of the average reward and outage probabilities for A2C agent.

it. This is translated into updating the Q-network in the case of DQN agent, and both Actor and Critic networks in the case of A2C agent.

## V. PERFORMANCE EVALUATION

In this section, we provide the performance evaluation of the proposed reinforcement learning approach. The simulation environment is implemented using python. We considered a carrier frequency  $f_c$  of 2 GHz, a noise variance  $N_0$  of  $-130$  dBm [18], and a Nakamai parameter  $m = 2$ . Furthermore, in order to reduce the action space we limit detected area for UAVs to a region of  $500m \times 500m$  and of 4 BSs and 20 UAVs in total. As for the DNN, we used Pytorch 1.7.1 [19]. For the hyper-parameters, we considered a learning rate of 0.003 for the two agents.

We evaluated the proposed DRL-based approach for traffic steering considering DQN and A2C algorithms in terms of the achieved outage probabilities. The obtained results are respectively depicted in Fig. 4 and Fig. 5. These evaluations have been performed considering 5000 episodes and a varied number of MNOs. In terms of the outage probabilities, we have compared the achieved results using DRL-approach against the initial deployments consisting of a grid topology and uniform distributions of MNOs on the UAVs (plotted in an

orange color (the more light color in black-and-white printed form) in Figs. 4 and 5).

As we can see, both agents are able to learn optimal solutions in selecting the MNOs to be used to steer the connection for UAVs. The obtained results show that the agents are able to increase the reward function, which is translated into reduced outage probabilities as shown in Figs. 4 and 5 (note that the reward function is the inverse of the outage probability, as provided in Equation (13)). Furthermore, the achieved reward increases with the number of considered MNOs. This observation is valid for the two agents. Indeed, the more MNOs are available, the more choices are present to distribute the MNOs on the UAVs in a way to achieve more enhanced spectral efficiency. The evaluation also shows that A2C agent achieves better results compared to DQN agent, especially when the number of MNOs increases. Indeed, increasing the number of MNOs is translated into increasing the action space. In this regard, the A2C agent demonstrates its effectiveness in supporting a large action space compared to the DQN agent. Compared to the initial assignment, the outage probabilities have been reduced by 4.13%, 4.08%, and 3.56% when considering the DQN agent on a deployment of 2 MNOs, 3 MNOs and 4 MNOs, respectively. It has also been reduced by 8.04%, 8.31%, and 11.13% when considering the A2C agent on a deployment of 2 MNOs, 3 MNOs and 4

MNOs, respectively.

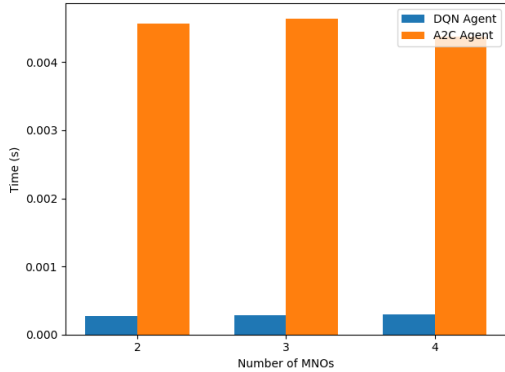


Fig. 6: Evaluation of the execution time for DQN and A2C agents.

We have also evaluated the execution time for selecting the target MNOs considering the two agents. The result of the evaluation is depicted in Fig. 6. This evaluation has been conducted by averaging 5000 trials. As we can see, the application of the proposed DRL approach involves short time which is less than 5 ms. On average, the execution time was 0.27 ms for the DQN agent and 4.5 ms for the A2C agent. This makes the DQN agent 16 times faster than the A2C agent. This is due to the fact that the A2C uses two networks during the selection, in addition to the application to a distribution function. Nevertheless, the execution time remains very short which is in the order of milliseconds. This demonstrates the applicability of such solutions for online use. We can also observe that increasing the MNOs, from 2 to 4, did not affect the execution time. In our implementation, increasing the number of MNO is translated into increasing the action space and the corresponding neural networks only at the output layer. We note that these evaluations have been performed on a x86\_64 machine with 8 CPUs of 2397.224 MHz, and recent studies have validated the deployment of RL algorithm for real UAVs.

## VI. CONCLUSION

This paper tackled the problem of connection steering in cellular UAVs. In particular, we focused on enabling quick and online decisions for selecting MNOs to be used for each UAV in a way to enhance the QoS in the network. To this end, the paper proposed an approach based on deep reinforcement learning. We considered two important RL algorithms, DQN and A2C. The simulation results showed that the two algorithms can learn optimal decisions in selecting the MNO to be used for steering the traffic. Remarkably, the results showed that the implemented A2C algorithm can achieve better results than the DQN algorithm, especially when the number of MNOs increases. On the other hand, while the execution time of the two algorithms is very short, the implemented DQN agent is faster than A2C agent.

## ACKNOWLEDGMENT

This work was partially supported by the European Union's Horizon 2020 Research and Innovation Program through the 5G!Drones Project under Grant No. 857031. It was also supported in part by the Academy of Finland 6Genesis project under Grant No. 318927.

## REFERENCES

- [1] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 899–922, Dec 2016.
- [2] O. E. Marai and T. Taleb, "Smooth and low latency video streaming for autonomous cars during handover," *IEEE Network*, vol. 34, no. 6, pp. 302–309, 2020.
- [3] H. Hantouti, N. Benamar, T. Taleb, and A. Laghrissi, "Traffic steering for service function chaining," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2018.
- [4] H. Hantouti, N. Benamar, and T. Taleb, "A novel compact header for traffic steering in service function chaining," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–6.
- [5] T. Taleb and A. Ksentini, "Vecos: A vehicular connection steering protocol," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 3, pp. 1171–1187, March 2015.
- [6] N. H. Motlagh, M. Bagaa, T. Taleb, and J. Song, "Connection steering mechanism between mobile networks for reliable uav's iot platform," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.
- [7] H. Hellaoui, A. Chelli, M. Bagaa, and T. Taleb, "Efficient steering mechanism for mobile network-enabled uavs," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.
- [8] U. Challita, W. Saad, and C. Bettstetter, "Interference management for cellular-connected uavs: A deep reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2125–2140, 2019.
- [9] H. Huang, Y. Yang, H. Wang, Z. Ding, H. Sari, and F. Adachi, "Deep reinforcement learning for uav navigation through massive mimo technique," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 1117–1121, 2020.
- [10] S. Fu, Y. Tang, Y. Wu, N. Zhang, H. Gu, C. Chen, and M. Liu, "Energy-efficient uav enabled data collection via wireless charging: A reinforcement learning approach," *IEEE Internet of Things Journal*, pp. 1–1, 2021.
- [11] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and A. Nallanathan, "Deep reinforcement learning based dynamic trajectory control for uav-assisted mobile edge computing," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2021.
- [12] S. Yin, S. Zhao, Y. Zhao, and F. R. Yu, "Intelligent trajectory design in uav-aided communications with reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8227–8231, 2019.
- [13] 3GPP, "Study on enhanced LTE support for aerial vehicles," *Technical Report, 3GPP TR 36.777*, 2017.
- [14] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Courier Corporation, 1964, vol. 55.
- [15] H. Hellaoui, A. Chelli, M. Bagaa, and T. Taleb, "Towards Mitigating the Impact of UAVs on Cellular Communications," in *2018 IEEE Global Communications Conference (GLOBECOM 2018)*, Dec. 2018.
- [16] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski et al., "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [17] V. R. Konda and J. N. Tsitsiklis, "Actor-critic algorithms," in *Advances in neural information processing systems*. Citeseer, 2000, pp. 1008–1014.
- [18] A. F. Molisch, *Wireless Communications*. Chichester: John Wiley & Sons, 2005.
- [19] "PyTorch pytorch website," <https://pytorch.org/>, accessed: 2021-03-30.