IEEE Open Journal of Vehicular Technology

Received 14 January 2020; accepted 26 January 2020. Date of publication 4 February 2020;
date of current version 26 February 2020. The review of this paper was arranged by Associate Editor N. Cheng.

Digital Object Identifier 10.1109/OJVT.2020.2971594

A Reinforcement Learning Approach for Fair User Coverage Using UAV Mounted Base Stations Under Energy Constraints

HASINI VIRANGA ABEYWICKRAMA[1] (Student Member, IEEE), YING HE[1] (Member, IEEE), ERYK DUTKIEWICZ[1] (Senior Member, IEEE), BEESHANGA ABEWARDANA JAYAWICKRAMA[2] (Member, IEEE), AND MARKUS MUECK[3] (Member, IEEE)

(Invited Paper)

[1]Global Big Data Technology Centre, University of Technology Sydney, Ultimo, NSW 2007, Australia
[2]Ericsson 164 40, Kista, Sweden
[3]Intel Mobile Communications, Neubiber 85579, Germany

CORRESPONDING AUTHOR: HASINI ABEYWICKRAMA (e-mail: hasini.v.abeywickrama@student.uts.edu.au)

This work was supported in part by Intel's University Research Office.

ABSTRACT Unmanned Aerial Vehicles (UAVs) are gaining popularity in many aspects of wireless communication systems. UAV-mounted mobile base stations (UAV-BSs) are an effective and cost-efficient solution for providing wireless connectivity where fixed infrastructure is not available or destroyed. However, UAV-BSs have their limitations and complications, for instance, limited available energy. In addition, when several UAV-BSs are deployed to provide coverage to a specific area, the possibility of inter-UAV collisions and the interference to ground users increase. We propose Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) based methods to deploy UAV-BSs under energy constraints to provide efficient and fair coverage to the ground users, while minimising inter-UAV collisions and interference to ground users. The proposed methods outperform the baseline methods by an average increase of 38.94% in system fairness, 42.54% in individual user coverage, and 15.04% in total system coverage, in comparison with the baseline methods.

INDEX TERMS Unmanned aerial vehicles (UAVs), wireless coverage, reinforcement learning.

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) are fast becoming a popular choice in a wide variety of applications in wireless systems. Their many desirable features, including cost efficiency, high manoeuvrability, and ease of deployment, make UAVs, a promising candidate to be used as mobile aerial base stations (BSs). UAV-mounted mobile base stations (UAV-BSs) can be deployed to provide wireless connectivity in areas of urgent need without infrastructure coverage, such as disaster-struck areas [1]. The ability of a UAV-BS to be sent to a specific target location immediately without having to deploy any infrastructure is one of the most significant advantages of UAV-BSs [2]. Unlike terrestrial BSs and even those mounted on ground vehicles, UAV-BSs can be deployed in any location and move along any trajectory constrained only by their aeronautical characteristics [1]. In addition, due to their high altitude, UAV-BSs have a higher chance of Line-of-Sight (LoS) connection with the ground users in comparison with the ground BSs [3].

However, UAV-BSs have their inherent limitations. Due to the very limited communication range, a large number of UAV-BSs would be required to provide wireless coverage to a large geographical area throughout the considered duration. This might not be possible due to the relatively high cost [3]. Thus, one or a limited number of UAV-BSs need to be deployed to provide wireless coverage to a large geographical area, and these UAV-BSs are required to fly and hover, providing wireless coverage to as many ground users in the considered area as possible. However, UAV-BSs have limited available energy, and they need to utilise the available

energy optimally in order to provide wireless coverage to as many users as possible and to prolong the network lifetime. In addition, the fairness of the system when providing coverage to the ground users needs to be considered. The probability of a small subset of users being provided with wireless coverage throughout the entire duration should be reduced. Additional complications arise when multiple UAV-BSs are deployed to serve a specific geographical area. Multiple UAVs sharing a common air space, flying in close proximity to each other, increase the possibility of inter-UAV collisions [4]. Furthermore, due to the same reasons, the probability of interference to ground users from the neighbouring cells (UAV-BSs) increases considerably.

The problem of optimising the paths of UAV-BSs to provide wireless coverage to a maximum number of ground users is a complicated problem with several concerns and limitations that are mentioned above. The complicated nature of the problem makes it hard to be solved using traditional optimisation techniques. Due to this reason, we propose using Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) techniques to solve this problem. With the proposed approaches, we aim to achieve the objectives below.

- Increase the total user coverage at the exhaustion of energy.
- Increase the number of individual users covered.
- Increase system fairness in providing coverage.
- Reduce collisions between UAV-BSs.
- Reduce interference to ground users caused by neighbouring UAV-BSs.

The main contributions of this piece work are summarised below.

- We propose RL and DRL based solutions for the problem of deploying UAV-BS(s) to provide wireless coverage to the ground users, considering fairness in providing coverage under energy constraints of the UAVs.
- We propose a Q-Learning (QL) based approach for the scenario of using a single UAV-BS and a Deep Q-Learning (DQL) approach for the scenario of using a set of UAV-BSs to provide coverage.
- We conduct extensive simulations to prove that the proposed methods outperform the baseline methods. In addition to increased fairness and user coverage the proposed DRL based method minimises the collisions between UAV-BSs and interference to ground users.

The rest of the paper is structured as below. Section II presents an overview of related work that is present in the current literature. Section III introduces our system model. Section IV presents a brief introduction to RL. Section V presents the proposed method. Section VI presents the simulation results and analysis, while Section VII concludes the paper.

II. RELATED WORK

A large volume of research has been conducted in recent years on UAVs in different areas. An extensive study on the use of UAVs in search and rescue, construction, coverage, and transportation was conducted from a communications point of view in [5]. A significant amount of research has been done in the areas of UAV trajectory optimisation [6]–[8], collision avoidance [9]–[11], and energy efficiency [12]–[14].

We discuss the recent work in literature that studied the use of a single UAV-BS, multiple UAV-BSs, and the use of Machine Learning (ML), Deep Learning (DL), RL, and DRL for UAV applications in wireless communication systems below.

A. DEPLOYMENT OF A SINGLE UAV-BS

Several studies have been carried out on deploying a single UAV-BS to assist in providing wireless coverage to ground users. The authors of [15] derived the optimal altitude for a UAV-BS to provide the maximum radio coverage to ground users. The authors of [16] studied the problem of 3D placement of a UAV-BS considering the traffic requirements and the density of ground users. In the letter [17], the authors modelled the UAV-BS placement in the horizontal dimension as a circle placement problem and a smallest enclosing circle problem. They proposed an efficient UAV-BS 3D placement method that maximised the number of covered ground users using the minimum required transmit power. The work presented in [18] aimed to maximise the number of covered users demanding different QoS requirements. The authors proposed using exhaustive search over a one-dimensional parameter in a closed region to obtain the optimal 3D location of the UAV-BS.

However, deploying coordinated multiple UAVs can perform tasks that go beyond the capabilities of a single UAV. This domain has not been explored in the above-cited work.

B. DEPLOYMENT OF MULTIPLE UAV-BSS

A UAV-BSs placement optimisation method called 'Spiral Method' was proposed in [1], which aimed to minimise the number of UAV-BSs needed to provide coverage to ground users. The proposed method placed UAV-BSs sequentially along the path connecting the extreme points of the convex hull of the uncovered ground users. The authors of [19] proposed a new network architecture called 'Coordinated multipoint (CoMP) in the sky' that mitigated the inter-user interference by the corporation gain of CoMP and used the mobility of UAVs to provide strong channel gains to mobile ground users. A method of dynamically clustering roaming users and energy efficient trajectory planning for a set of UAV-BSs to serve the user clusters was presented in [20]. A UAV-BSs deployment and mobility control algorithm was proposed in [21]. The UAV-BSs were made to fly in a macro-hotspot continually, based on a game theory based mobility algorithm, continuously serving the mobile ground users. The authors of [22] studied the problem of deploying a set of heterogeneous UAV-BSs with different flying speeds, operating altitudes, and coverage radii, minimising the maximum deployment delay among all UAV-BSs and minimising the total deployment delay. The authors of [23] proposed deploying a set of UAV-BSs in a way that optimised the quality of coverage. They focused on minimising the average Euclidean

distance between users and the nearest UAV-BS. They proposed a recursive algorithm that moved UAV-BSs closer to the centres of the corresponding Voronoi cells at each iteration.

However, energy efficiency and fairness in providing coverage to ground users have not been considered in most of the above-cited work.

C. MACHINE LEARNING FOR UAVS IN WIRELESS COMMUNICATION APPLICATIONS

ML has attracted the attention of the researchers due to its ability of solving problems that are impossible or too difficult to be solved using traditional methods. ML, RL, and DL have been widely used in recent years for UAV applications in wireless communications systems.

The key use cases of cellular-connected UAVs in UAV based delivery systems, real-time multimedia streaming applications, intelligent transportation systems were discussed in [24], along with the main wireless and security challenges faced and possible Artificial Neural Network (ANN) based solutions. The authors of [25] proposed a space-air-ground IoT network for offloading computation-intensive applications where they employed a policy gradient based actor-critic learning algorithm to determine the optimal offloading policy. A QL based approach was proposed in [26] to minimise inter-cell interference and save energy for ultradense small cells. Further, a DQL based approach was proposed to accelerate the learning speed for the ultra-dense small cells with a large number of active users. The authors of [3] proposed a centralised method named DRL-based 'Energy-efficient Control for Coverage and Connectivity,' which ensured effective and fair coverage to ground users. The actor-critic method Deep Deterministic Policy Gradient (DDPG) was used as the base for the proposed method. The above proposed method was extended to a distributed DRL-based control solution in [27]. In [28], a threefold solution was proposed for using UAV-BSs to provide coverage to ground users. First, the authors proposed using genetic algorithm based K-means clustering (GAK-means) to partition the users into cells. Second, a QL based algorithm was proposed for static 3D deployment of the UAV-BSs. Finally, a QL based movement algorithm was proposed for the UAV-BSs to serve the roaming ground users.

However, the practical scenario of maximising coverage at the exhaustion of available energy of the UAV-BSs is not considered in any of the above cited work. Additionally, minimising interference to ground users and collisions between UAV-BSs have not been taken into consideration in work present in literature.

In our work presented in this paper, we consider several factors that have not been widely considered in literature when using UAV-BSs. We aim to maximise coverage at the exhaustion of available energy in the UAVs, which is a limitation that has not been considered in most of the work in literature. We aim to minimise inter-UAV collisions and the interference to the ground users caused by neighbouring UAV-BSs as well, which has been often overlooked in the literature.

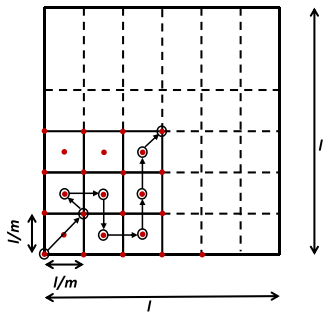


FIGURE 1. Virtually Discretised Area.

III. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a square area of width l, with a set $\mathcal{N}$ of N ground users uniformly and at random positioned in the considered area. We initially consider using one UAV-BS U to provide downlink communication to the ground users. We assume that the UAV-BS is backhaul-connected via satellite links. It is understood that satellite-UAV links are not commercialised yet. However, UAV-BSs being backhaul-connected through satellite links, is a common assumption in literature [1], [20], and [28]. Later, we extend the scenario to use a fleet of UAV-BSs to provide downlink communication to ground users in the said area.

The UAV-BS U, has a limited amount of energy E available for flying (E is expected to be used only for manoeuvring purposes, not for transmission. However, the energy required for communication purposes is negligible in comparison to the energy required for a UAV to fly and hover [29]). The UAV-BS should utilise the available energy to provide fair coverage to the maximum number of ground users.

The 2D coordinates of each ground user $n \in \mathcal{N}$ is known and given by (x_n, y_n). The 3D coordinates of the UAV-BS at time t is given by (x_t, y_t, h). The UAV-BS is assumed to fly in a fixed altitude h. According to [30] 10 m is the optimal height for positioning a typical small cell BS. Lowering the antenna below 10 m would cause possible coverage issues, and an antenna height higher than 10 m would increase interference with neighbouring cells [2]. Thus, we assume the UAV-BS to fly and hover at a fixed altitude of 10 m.

Assuming the UAV-BS's coverage radius is $l/2\,m$, for exposition purposes, we discretise the considered area into a set of unit sized square areas with a side length of l/m. Thus, the area is virtually divided into $m \times m$ unit squares. The UAV-BS is assumed to station (hover) only at the corners and centres of the said set of unit squares shown by the red dots in Fig. 1. In literature, it is common practice to assume that the entire cell area is being covered if the UAV-BS is positioned at the centre of the cell [3]. For high user density scenarios, this assumption is unfair, as $25(4-\pi)\%$ of the entire area is not being covered the entire period of time (Fig. 2). Considering the corners and the centres of the cells to be possible hovering points eliminates this unfairness. The number of possible hovering points is $M = 2m(m+1)+1$. The possible hovering points

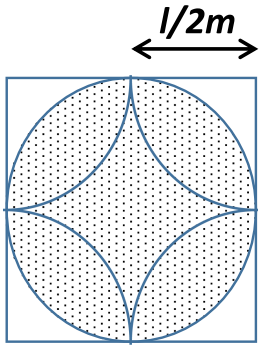


FIGURE 2. Coverage Based on Hovering Point.

for the UAV-BS are hereon denoted by the set $\mathcal{M}$, which has a cardinality of M.

A. CHANNEL MODEL

We adopt the air-to-ground (ATG) channel model with probabilistic LoS and NLoS connections proposed in [15]. The proposed probability of LoS connection between a ground user and the UAV-BS is given by the below equation.

$$P_{LOS}(h, r) = \frac{1}{1 + g\,exp(-z(\theta - g))} \tag{1}$$

where P_{LOS} is the probability of LoS connection, h is the relative flying altitude of the UAV-BS, r is the distance between the ground user and the UAV's location projected on ground, g and z are statistical parameters that depend on the environment. θ is $arctan(h/r)$ in degrees. Based on the basic theories of probability, the probability of having a non line of sight (NLoS) connection is $P_{NLOS}(h, r) = 1 - P_{LOS}(h, r)$.

We assume that the Doppler effect due to the mobility of the UAV-BS is compensated for based on existing techniques (eg. frequency synchronisation using a phase-locked loop [31]) as done in [32].

The path loss (L) can be calculated by,

$$L = 20\log\left(\frac{4\pi f_c d}{c}\right) + \eta \tag{2}$$

where f_c is the carrier frequency and c is the speed of light. d is the distance between the ground user and the UAV-BS ($d = \sqrt{h^2 + r^2}$). η is the mean additional loss, which would take different values for LoS (η_{LoS}) and NLoS (η_{NLoS}) scenarios [15].

ATG communication is dominated by LoS connections. However, the obstacles in a realistic environment, such as buildings and trees might disturb LoS. Thus, we consider the probabilistic average path loss, averaged over the LoS and NLoS scenarios. The path loss between a UAV-BS hovering at relative altitude h and a user r away from UAV-BS's ground projection can be calculated as below.

$$\begin{aligned} L_{h,r} &= L_{LoS} \times P_{LoS} + L_{NLoS} \times P_{NLoS} \\ &= P_{LoS}(\eta_{LoS} - \eta_{NLoS}) + L_{NLoS} \\ &= \frac{\eta_{Los} - \eta_{NLoS}}{1 + g\,exp(-z(\theta - g))} \\ &\quad + 20\log\left(\frac{r}{\cos\theta}\right) + 20\log\left(\frac{4\pi f_c}{c}\right) + \eta_{NLoS} \end{aligned} \tag{3}$$

We assume that the transmit power (P_{tx}) of the UAV-BS and carrier frequency (f_c) are fixed and the maximum allowed path loss ($PL_{\max}$) at the receiver for reliable communication is given. According to (3), for fixed f_c the path loss depends on r and θ. Since the UAV-BSs are assumed to fly at a fixed altitude h, and $\tan\theta = h/r$, the path loss threshold can be considered a coverage disk of radius R', as all receivers inside this coverage disk would have a path loss less than or equal to $PL_{\max}$.

B. PROBLEM FORMULATION

We aim to find the optimal path for the UAV-BS subjected to its available energy constraints in a way that maximises

- Total coverage of the ground users.
- Fairness in providing coverage.

The trajectory of the UAV-BS can be thought of as a directed graph $G = (V, L)$, V is the set of vertices ($V \subset \mathcal{M}$) which represents the hovering points of the UAV-BS, shown by the red dots with a black outline in Fig. 1. L is the set of edges which represents the flight path of the UAV-BS, shown by the black arrows in Fig. 1. Thus the path of the UAV-BS can be represented by $P = v_0, v_1, \ldots, v_t (v_i \in V, \forall i)$. We send control commands to update the UAV-BS's position every t' seconds. t' comprises of the time taken by the UAV-BS to fly between two hovering points t_f and the hovering time t_c. The UAV-BS is assumed to provide coverage to the users only when it is hovering.

Since most of the real world UAV application scenarios are per instruction based [33], [34], we assume our system behave per instruction. Instructions are sent to the UAV-BSs every t' seconds and the UAV-BSs continue following an instruction until a new instruction is received.

1) TOTAL COVERAGE OF THE GROUND USERS

We aim to provide coverage to the maximum number of users with the available energy. To measure the user coverage provided by the UAV-BS, we introduce a 'coverage track' (similar to the 'coverage score' presented in [3]. However, we track the coverage of users not specific geographical points) for each user $n \in \mathcal{N}$. The coverage track of user n at time step t, (Cov_n^t) is determined as below,

$$Cov_n^t = \begin{cases} 1, & \sqrt{(x_n - x_t)^2 + (y_n - y_t)^2} \leq R' \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

The total coverage of the n^{th} ground user at time interval t is:

$$Cov_n(t) = \sum_{i=0}^{t} Cov_n^i \tag{5}$$

The total coverage of the entire user set at time interval t, $Cov(t)$, is give by (6)

$$Cov(t) = \sum_{n=1}^{N} \sum_{i=0}^{t} Cov_n^i \tag{6}$$

2) FAIRNESS IN PROVIDING COVERAGE

Fairness can be considered both at the system level and individual level. The system fairness addresses the overall fairness amongst all individuals in the system, while individual fairness indicates whether a certain individual is treated fairly by the system [35]. Our objective is to increase system fairness.

It is a possibility that in most timeslots, a small subset of users are provided with network coverage while the majority of the users are left uncovered. This leads to unfair coverage. We use a widely used matrix for fairness, Jain's fairness index [36] as a means of measuring fair coverage. The fairness index of the system at time interval t, F_t is given by,

$$F_t = \frac{\left(\sum_{n=1}^{N} \sum_{i=0}^{t} Cov_n^i\right)^2}{N\left(\sum_{n=1}^{N} \left(\sum_{i=0}^{t} (Cov_n^i)^2\right)\right)} \tag{7}$$

3) UAV-BS OPTIMAL PATH FINDING PROBLEM

For ease of exposition we define a variable $\alpha_{a,b}$ which indicates whether the UAV-BS would fly directly between the hovering points a and b. $\alpha_{a,b} \in \{0, 1\}$. $\alpha_{a,b} = 1$ if the UAV-BS would fly from point a to point b, $(a, b \in \mathcal{M})$ and 0 otherwise. In order to determine the optimal path for the UAV-BS, we propose solving the optimisation problem given below.

$$\begin{aligned} \max \quad & \sum_{n=1}^{N} \sum_{i=0}^{T} Cov_n^i \\ \text{subject to} \quad & c1. \sum_{a=1}^{M} \sum_{b=1, b\neq a}^{M} \alpha_{a,b}\, e_{a,b} \leq E \\ & c2.\ F_t < F_{t+1}, \quad (\text{if } F_t \neq 1 \ \ \forall t \leq T) \\ & c3.\ \alpha_{a,b} t_{a,b} < t' \end{aligned} \tag{8}$$

The objective function in (8) aims to maximise the total user coverage of the system, which can be derived from (4)–(6). The constraint $c1$ makes sure the energy spent on manoeuvring the UAV-BS is within the limit of available energy. $e_{a,b}$ is the energy required to fly between the points a and b. We follow the energy model proposed in [29] for energy predictions.

The constraint $c2$ assures that the UAV-BS always positions itself in a way that increases system fairness. This constraint makes sure that the UAV-BS does not always provide coverage to the same subset of users. However, this constraint is only checked if $F_t \neq 1$. Since 1 is the maximum possible value for the fairness index, when the system reaches the fairness index 1, no matter what action the UAV-BS takes (hover in the current position or move to a new position), the fairness would reduce, unless the entire set of ground users is covered by taking the action, which is highly unlikely. Thus, the second constraint is checked only if the system fairness has not reached the maximum.

$t_{a,b}$ represents the time needed to travel between the points a and b. The constraint $c3$ makes sure that the time needed for the UAV-BS to fly between two points is less than the time interval between two control commands.

The considered square area can be thought of as a complete graph (assuming the speed limitations of the UAV-BS would not constraint the flight between the two furthest points within the allowed time) with possible hovering points as vertices and the paths to the hovering points as edges. This reduces to a complete graph with a cost on each edge (energy required) and a reward at each vertex (number of users that can be covered at each point). The objective is to find a path that maximises the reward within a given cost. Thus, if we relax the constraints in the optimisation problem (8), it falls to the form of Traveling Purchaser Problem (TPP), which is a generalisation of the classical Traveling Salesman Problem (TSP) [37]. TSP and thus TPP are NP-hard, making the problem in (8) NP-hard as well.

The energy model presented in [29], shows that the energy requirement for a trajectory is not linear, making the constraint $c1$ non-linear.

The condition to satisfy the fairness constraint $c2$ can be derived as below.

If we consider t time slots, a user can be covered for 0 to t number of time slots. It is assumed that in each of above time slots, $m_0, m_1, m_2, \ldots, m_t$ users are provided coverage. Thus, from (7), the fairness of the system at t can be expressed as below.

$$F_t = \frac{\left(\sum_{i=0}^{t} i m_i\right)^2}{N \sum_{i=0}^{t} i^2 m_i} \tag{9}$$

For clarity, if $A = \sum_{i=0}^{t} i m_i$ and $B = \sum_{i=0}^{t} i^2 m_i$, then, $F_t = A^2/NB$.

To satisfy $(c2)$,

$$\begin{aligned} \frac{A^2}{NB} &< \frac{(A + (t+1)m_{t+1})^2}{N(B + (t+1)^2 m_{t+1})} \\ m_{t+1} &> \frac{A(A(t+1) - 2B)}{B(t+1)} \end{aligned} \tag{10}$$

The above inequality is for the number of users covered. However, when measuring fairness, the individuality of the users needs to be considered as well. This makes the constraint $c2$ non-trivial to be satisfied. Due to these reasons, the problem (8), is too difficult to be solved using traditional optimisation techniques.

We propose using an RL approach to solve this problem, specifically based on QL.

IV. REINFORCEMENT LEARNING AND Q-LEARNING BASICS

In RL, an agent constantly interacts with an environment, typically formulated as a Markov Decision Process (MDP). The system consists of a set $\mathcal{S}$ of states and a set $\mathcal{A}$ of possible actions for each state. The agent interacts with the environment in discrete time intervals. At each time interval t, the agent observes the system state ($s_t \in \mathcal{S}$), and performs an action ($a_t \in \mathcal{A}$) that changes the system state to ($s_{t+1} \in \mathcal{S}$). Based on s_t, a_t, s_{t+1}, the agent receives a numerical reward r_t. The objective of RL is to find the optimal policy $\pi(s, a)$, that maps a state to an action that maximises the discounted cumulative reward $R = \sum_{t=0}^{\infty} \gamma r_t$, where γ, $(0 \leq \gamma \leq 1)$ is the discount factor which determines the importance given to future rewards.

QL is a widely used model-free algorithm for RL first proposed by Watkins and further developed in 1992 [38]. The Q function calculates the quality of a state-action combination:

$$Q : S \times A \rightarrow \mathbf{R} \tag{11}$$

The QL update rule is given by the below equation.

$$Q_{t+1}(s_t, a_t) = (1 - \alpha) Q_t(s_t, a_t) + \alpha \left(r_t + \gamma \max_a Q(s_{t+1}, a) \right) \tag{12}$$

where α is the learning rate and all the other symbols have the same meanings as described above.

QL becomes highly inefficient in scenarios where the state space is large. To overcome this limitation QL can be combined with function approximation. In Deep Q-Learning (DQL) artificial deep neural networks (DNN) are used as the approximator for the Q-function.

V. PROPOSED REINFORCEMENT LEARNING BASED APPROACH

In the proposed RL approach, an agent periodically inspects the status of the environment, chooses the best action based on the current state and informs the UAV-BS of the action to be taken.

A. STATE SPACE AND ACTION SPACE

The state s_t, at each time epoch t consists of following information.

- total coverage of each ground user until the current time interval t, $Cov_n(t) \in [0, t] (\forall n \in \mathcal{N})$.
- current position of the UAV-BS $v_t = (x_t, y_t)$.
- available energy of the UAV-BS $e_t \in [0, E]$.

With the above information included, the format of the state would be $s_t = [Cov_1(t), \ldots, Cov_n(t), v_t, e_t]$, with a cardinality of $(N + 2)$.

Theoretically, a UAV can fly in any direction, and this leads to a continuous action space. This might result in impractically high training time for the network and possible divergence in QL. Thus, we aim to reduce the action space. This is achieved by restricting the possible actions the UAV-BS can take at each state s_t. Hence, we assume that the UAV-BS can fly in a direction given by $A = \{0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/4, 2\pi\}$ to the next immediate hovering point in selected direction. 0 indicates hovering in the current position. We further assume UAV-BSs always fly at the same constant speed, ensuring that UAV speed would always satisfy the constraint shown in $c3$.

Based on the action taken, the next position of the UAV-BS v_{t+1} can be calculated with respect to the current position v_t as below.

$$v_{t+1} = \begin{cases} v_t, & a_t = 0 \\ v_t + (l/2\,m, l/2\,m), & a_t = \pi/4 \\ v_t + (l/m, 0), & a_t = \pi/2 \\ v_t + (l/2\,m, -l/2\,m), & a_t = 3\pi/4 \\ v_t + (0, -l/m), & a_t = \pi \\ v_t + (-l/2\,m, -l/2\,m), & a_t = 5\pi/4 \\ v_t + (-l/m, 0), & a_t = 3\pi/2 \\ v_t + (-l/2\,m, l/2\,m), & a_t = 7\pi/4 \\ v_t + (0, l/m), & a_t = 2\pi \end{cases} \tag{13}$$

TABLE 1. Description of Parameters Used

Parameter	Description
l	Width of the considered area
N	Number of ground users
E	Energy available for flying
K	Number of UAV-BSs
(x_n, y_n)	2D coordinates of the n^{th} ground user
(x_t^k, y_t^k)	2D coordinates of the k^{th} UAV-BS at time iteration t (for one UAV-BS scenario k has no effect)
h	UAV-BS flying altitude
v_t^k	Position of the $k^{th} UAV-BS$ at time iteration t (for one UAV-BS scenario k has no effect)
$m \times m$	Number of unit areas
M	Number of possible hovering points
R'	Coverage radius of the UAV-BS
Cov_n^t	Instantaneous coverage track of user n at time iteration t
$Cov_n(t)$	Total coverage of user n at time iteration t
$Cov(t)$	Total coverage of entire user set at time iteration t
F_t	System fairness at time t
t_{ab}	Flying time between points a and b
$\alpha_{a,b}$	UAV-BS path indicator from a and b
e_{ab}	Energy required to fly between points a and b
S, A	Set of states and actions
R	Discounted cumulative reward
s_t, a_t, r_t	State, action and reward at time epoch t
γ	Discount factor
α	Learning rate

B. REWARD FUNCTION

At each time epoch t, the reward is calculated based on the reward function below (The initial system fairness, $F_0 = 0$).

$$r_t = \begin{cases} \sum_{n=1}^{N} Cov_n^t \times \left(F_t - F_{(t-1)}\right), & F_{(t-1)} \neq 1 \\ \sum_{n=1}^{N} Cov_n^t \times \left(F_t\right), & \text{otherwise} \end{cases} \tag{14}$$

We propose taking the difference in system fairness into account when calculating the reward, aiming to prompt the UAV-BS to move in a manner the system fairness is increased with each step (if the system fairness has not reached the maximum of 1 yet). The authors in [3] and [27], consider the instantaneous fairness of the system at each time step when calculating the reward as opposed to considering the difference in fairness that is proposed in this paper. However, considering the instantaneous fairness of the system tends to make the UAV-BS reluctant to fly. A comparison of the system performance for the two reward functions is presented in Section VI.

The proposed algorithm for training the system is given in Algorithm 1. We start the training process by arbitrarily initialising the Q-Table (Q(s, a)). We employ an ε-greedy policy [38] to determine the action to be taken at each state. We select action a_t either randomly with a probability of ε (explore) or select the action that results in the highest Q value for the current state otherwise (exploit). Based on the resulting state s_{t+1}, we calculate the reward r_t according to (14). Since we are considering a closed area in this scenario, the UAV-BS has to respect the area boundaries. To impose this, we add a penalty p to the reward every time an action would result in the UAV-BS going over the area boundaries. The conditions for applying the penalty p is given in (15).

$$p = \begin{cases} \lambda_1, & 0 > x_t > l \quad \text{or} \quad 0 > y_t > l \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

C. MULTI-UAV SCENARIO

Next, we consider the scenario of deploying multiple UAV-BSs to provide coverage to a geographical area. We assume a set $\mathcal{K}$ of K UAV-BSs is deployed to provide coverage to the ground users. We assume K is not sufficient to provide coverage to all the users throughout the entire duration, thus the UAV-BSs need to fly and hover at different points to provide fair coverage to the ground users in the considered area.

Multiple UAV-BSs make the system more complex and increase the state space of the problem. A large state space increases the size of the Q-table and the time taken to look up. This makes using QL highly inefficient in solving the problem. Thus, we propose using DQL to solve the scenario of multiple UAV-BSs, where a neural network is used to approximate the Q-value function.

Having multiple UAV-BSs introduce complexities that are not present in the single UAV-BS scenario. Accordingly we have to take the below facts into consideration in addition to the factors considered in the single UAV-BS scenario.

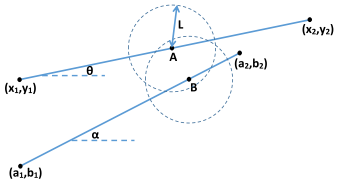


FIGURE 3. Collision Between Two UAV-BSs.

Algorithm 1: Proposed Method With Q-Learning.

Initialise Q(s, a) arbitrarily
for episode:= 1, ..., M **do**
  Get the initial state s_1
  while available_energy > 0 **do**
    $a_t = \begin{cases} \text{random action}, & \text{with } \varepsilon \text{ probability} \\ \text{argmax}_a Q(s_t, a), & \text{otherwise} \end{cases}$
    Execute a_t and obtain s_{t+1}
    Calculate r_t based on (14)
    Calculate p based on (15)
    if $p > 0$ **then**
      $r_t = r_t - p$
      UAV-BS stays in the same place
      Update s_{t+1} accordingly
    end if
    $Q_{t+1}(s_t, a_t) =$
    $(1-\alpha)Q_t(s_t, a_t) + \alpha(r_t + \gamma \max_a Q(s_{t+1}, a))$
  end while
end for

- Collision avoidance between UAV-BSs.
- Reducing interference to ground users caused by neighbouring UAV-BSs.

1) COLLISION DETECTION

As commonly done in the literature ([10], [4]), a safety distance (L) is defined for the UAVs. When two or more UAVs are closer than the safety distance to each other, it is identified as a collision (Fig. 3).

A possible collision between two UAV-BSs is determined as below. If UAV-BS1 flies from point (x_1, y_1) to point (x_2, y_2), its path is given by the equation (16).

$$y = \frac{y_1 - y_2}{x_1 - x_2} x + \frac{x_1 y_1 - x_2 y_1}{x_1 - x_2} \tag{16}$$

Assuming $m_1 = \frac{y_1 - y_2}{x_1 - x_2}$ and $c_1 = \frac{x_1 y_1 - x_2 y_1}{x_1 - x_2}$, path of UAV-BS1 is reduced to $y = m_1 x + c_1$.

Similarly, if UAV-BS2 flies from point (a_1, b_1) to point (a_2, b_2), and $m_2 = \frac{b_1 - b_2}{a_1 - a_2}$ and $c_2 = \frac{a_1 b_1 - a_2 b_1}{a_1 - a_2}$, its path is given by $y = m_2 x + c_2$.

UAV-BS1 has a speed of v_1 and UAV-BS2 has a speed of v_2. At time t UAV-BS1 arrived at point A and UAV-BS1 arrived at point B. The coordinates of A and B can be determined as below.

$$A \equiv ((x_1 \pm v_1 t \cos\theta), (y_1 \pm v_1 t \sin\theta))$$
$$B \equiv ((a_1 \pm v_2 t \cos\alpha), (b_1 \pm v_2 t \sin\alpha)) \quad (17)$$

where $\theta = \tan^{-1} m_1$ and $\alpha = \tan^{-1} m_2$.

The exact values for A and B can be determined by considering the distances between the possible points and the target points. For ease of exposition, we consider $A \equiv ((x_1 + v_1 t \cos\theta), (y_1 + v_1 t \sin\theta))$ and $B \equiv ((a_1 + v_2 t \cos\alpha), (b_1 + v_2 t \sin\alpha))$.

For a collision to occur between UAV-BS1 and UAV-BS2,

$$((x_1 - a_1) + (v_1 \cos\theta - v_2 \cos\alpha)\, t)^2 +$$
$$((y_1 - b_1) + (v_1 \sin\theta - v_2 \sin\alpha)\, t)^2 \leqslant L^2$$
$$Ht^2 + It + J \leqslant 0 \quad (18)$$

where,

$$H = v_1^2 + v_2^2 - 2v_1 v_2\, (cos\theta \cos\alpha - \sin\theta \sin\alpha)$$
$$I = 2((x_1 - a_1)\,(v_1 \cos\theta - v_2 \cos\alpha)$$
$$+ (y_1 - b_1)\,(v_1 \sin\theta - v_2 \sin\alpha))$$
$$J = (x_1 - a_1)^2 + (y_1 - b_1)^2 - L^2 \quad (19)$$

Solving (18),

$$\left(t - \left(\frac{-I + \sqrt{I^2 - 4HJ}}{2H}\right)\right)\left(t - \left(\frac{-I - \sqrt{I^2 - 4HJ}}{2H}\right)\right) \leqslant 0 \quad (20)$$

For a collision to occur, a valid solution should be available for t, such that $0 < t \leq t'$.

In the proposed method, possible collisions are detected as mentioned above and UAV-BSs are prompted to move into hovering points that do not result in collisions. Collisions involving more than two UAV-BSs can be detected by breaking down the collision into multiple two-UAV collisions, as done in [10].

One solution for avoiding inter-UAV collisions is using the height separation technique (deploying UAVs at different heights). However, this leads to deploying UAV-BSs at a wide range of heights, which would cause performance degradation of the system [21]. Thus, we propose detecting the possible collisions and training the system to come up with UAV-BS paths that result in minimal possible collisions.

2) REDUCING INTERFERENCE

We aim to reduce the interference to the ground users by the neighbouring UAV-BSs. To measure the impact of interference, we introduce four levels of interference.

- Level 0: When the distance between two UAVs is $2\times$ diameter of the coverage disk (Fig. 4(a)). The interference to ground users is the lowest in this scenario.
- Level 1: When the distance between two UAVs is $\sqrt{2}\times$ diameter of the coverage disk (Fig. 4b). The interference to ground users is low in this scenario.

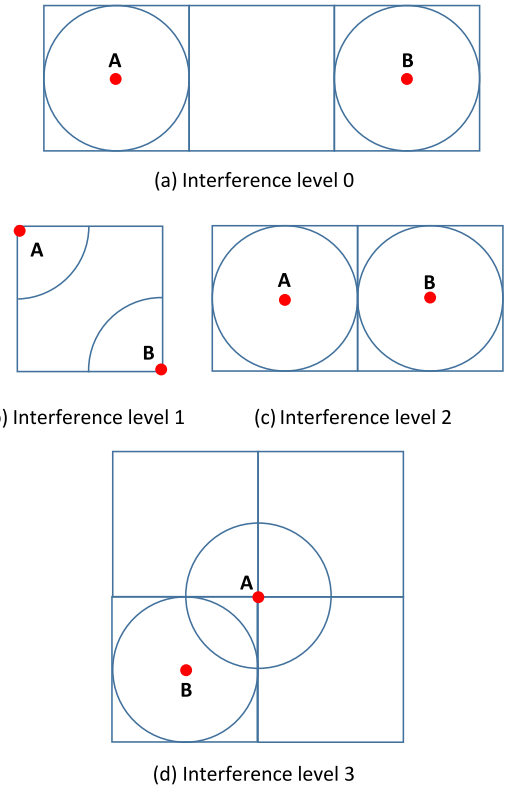


FIGURE 4. Interference levels based on UAV positions.

- Level 2: When the distance between two UAVs is equal to the diameter of the coverage disk (Fig. 4c). The interference to ground users is moderate in this scenario.
- Level 3: When the distance between two UAVs is smaller than the diameter of the coverage disk (Fig. 4d). The interference to ground users is high in this scenario.

To adapt to the scenario of multiple UAV-BSs, we have to change the state space $\mathcal{S}$ accordingly. The state s_t, at each time epoch t should consist of the following information.

- total coverage of each ground user until the current time interval t, $Cov_n(t) \in [0, t]$, $(\forall n \in \mathcal{N})$.
- current position of each UAV-BS $v_t^k = (x_t^k, y_t^k)$, $(\forall k \in \mathcal{K})$.
- available energy of each UAV-BS $e_t^k \in [0, E]$, $(\forall k \in \mathcal{K})$.

The format of s_t would be $s_t = [Cov_1(t), \ldots, Cov_n(t), v_t^1, \ldots, v_t^k, e_t^1, \ldots, e_t^k]$. The cardinality of the state is $(N + 2K)$.

In the training process, in addition to the penalty p introduced in (15), two additional penalties are introduced. The penalties are applied to each UAV-BS.

- Possibility of collisions between UAV-BSs (p_2).
- Possibility of causing interference to the users by the neighbouring UAV-BSs (p_3).

The penalties p_2 and p_3 for UAV-BS k is defined below.

$$p_2^k = \begin{cases} \lambda_2, & \text{if a solution for (20) exists} \\ 0, & \text{otherwise} \end{cases} \quad (21)$$

Algorithm 2: Proposed Method with Deep Q-Learning.

1: Initialise replay memory D into capacity B
2: Initialise action-value function Q with random weights θ
3: Initialise target action-value function $\hat{Q}$ with random weights $\theta^- = \theta$
4: **for** episode:= 1, ..., M **do**
5: Get the initial state s_1
6: **while** available_energy > 0 **do**
7: $p, p_2, p_3 = 0$
8: Select
$$a_t = \begin{cases} \text{random action}, & \varepsilon \text{ probability} \\ \text{argmax}_a Q(s_t, a; \theta), & \text{otherwise} \end{cases}$$
9: Execute a_t and obtain s_{t+1}
10: Calculate r_t based on (14)
11: **for** i:= 1, ..., K **do**
12: Calculate p^i based on (15)
13: $p = p + p^i$
14: **if** $p^i > 0$ **then**
15: UAV-BS i stays in the same place
16: Update s_{t+1} accordingly
17: **end if**
18: Calculate p_2^i based on (21)
19: $p_2 = p_2 + p_2^i$
20: Calculate p_3^i based on (22)
21: $p_3 = p_3 + p_3^i$
22: **end for**
23: $r_t = r_t - (p + p_2 + p_3)$
24: Store transition (s_t, a_t, r_t, s_{t+1}) in D
25: Sample random minibatch of transitions (s_t, a_t, r_t, s_{t+1}) from D
26: Set
$$y_j = \begin{cases} r_j, & \text{terminal} \\ r_j + \gamma \max_{a'} \hat{Q}(s_{t+1}, a'; \theta^-) & \text{otherwise} \end{cases}$$
27: Perform gradient descent step on $(y_j - Q(s_j, a_j; \theta))^2$ w.r.t the network parameter θ
28: Every C steps reset $\hat{Q} = Q$, i.e., set $\theta^- = \theta$
29: **end while**
30: **end for**

If the distance between UAV-BS k and UAV-BS q is d_{kq},

$$p_3^k = \begin{cases} \lambda_3^0, & d_{kq} = 2\,l/m \ \text{(Figure 4a)} \\ \lambda_3^1, & d_{kq} = \sqrt{2}l/m \ \text{(Figure 4b)} \\ \lambda_3^2, & d_{kq} = l/m \ \text{(Figure 4c)} \\ \lambda_3^3, & d_{kq} < l/m \ \text{(Figure 4d)} \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

Since the target is to maximise the reward, the system tries to find a trade-off between the penalties.

The proposed method of positioning UAV-BSs to provide coverage using DQL (inspired by [39]) is shown in Algorithm 2.

The basic idea of DL is to train the system to reach a target, and in traditional DL this target does not change. However, in RL as the agent explores the environment, the target continuously changes in each iteration. This results in unstable, most of the time non-converging training. In order to overcome this in DQL, two networks are used. One network is used as the function approximator and the other to estimate the target. The target network has the same architecture and the parameters as the function approximator. Information of the latest transactions are stored in memory (line 24 in Algorithm 2) a subset of which is used to train the network. After C steps, the parameters of the target network are replaced with that of function approximator (line 28 in Algorithm 2).

TABLE 2. Simulation Parameters

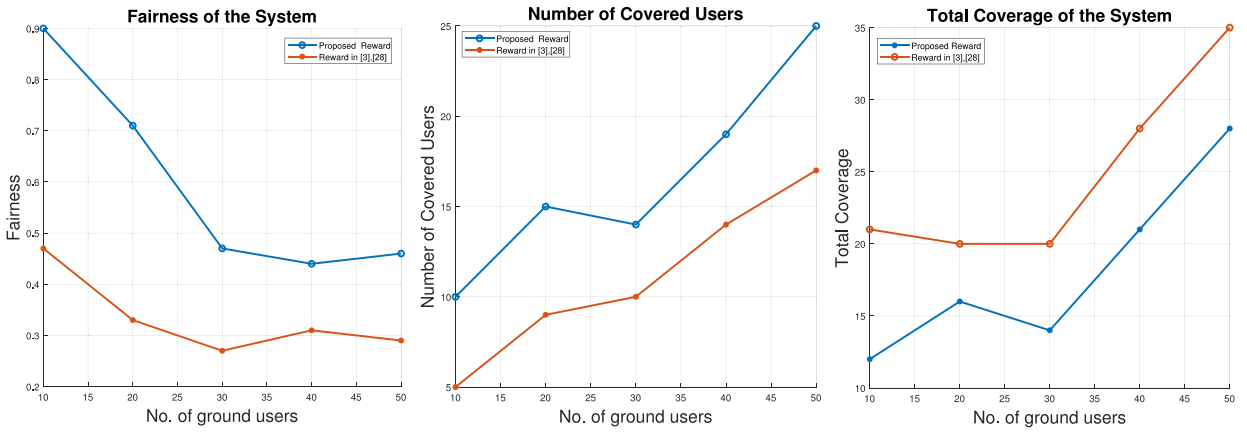
Parameter	Value
$l \times l$	$2 \times 2\ km^2$
N	60 - 100
E	2000 - 6000 units
K	1 - 5
R'	250 m
L	100 m
ϵ	0.7
α	0.01
γ	0.9
λ	10
λ_2	50
λ_3^0	5
λ_3^1	10
λ_3^2	15
λ_3^3	20

VI. SIMULATION RESULTS

Our simulation setup was developed using Tensorflow 1.14 and Python 3.6.5. We considered a target area of 2×2 km. The coverage radius of the UAV-BS was considered to be 250 m. The simulation parameters are summarised in Table 2.

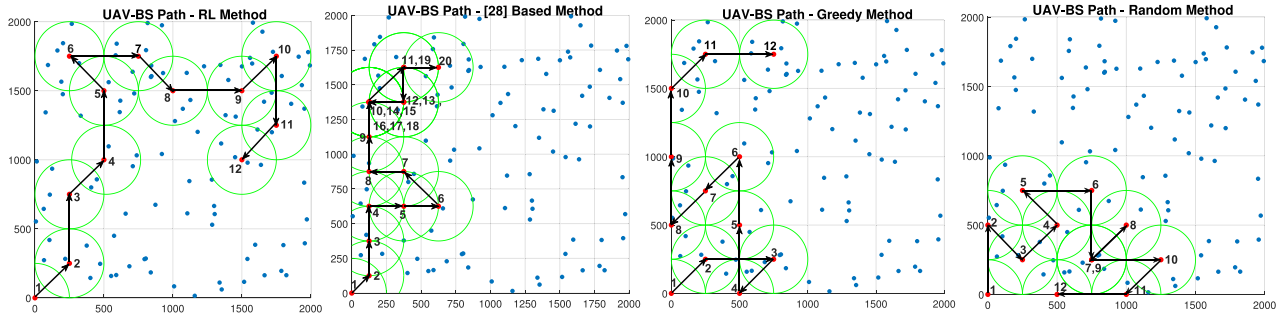
Initially, we checked the effectiveness of the reward function proposed in this paper in comparison to that proposed in [3] and [27]. The simulations were run with the two different reward functions, while keeping all other simulation parameters the same. The results are shown in Fig. 5. The simulation results show that the reward function proposed in this paper results in better total system fairness and a higher number of individual users covered while the reward function presented in [3] and [27] results in better total system coverage. According to (6), the total system coverage can be increased by providing coverage to a subset of users for a long time duration, however, with reduced system fairness. This is the reason for increased total system coverage and reduced fairness and individual coverage with the reward function presented in [3] and [27]. Since the reward function proposed in this paper prompts the UAV-BS to fly and provide coverage to as many individual users as possible, the system fairness and the total number of individual users covered are higher.

We evaluate the performance of the proposed algorithm based on the below parameters:



(a) System Fairness Comparison (b) Individual User Coverage Comparison (c) Total System Coverage Comparison

FIGURE 5. Comparison of system fairness, number of covered users and total coverage of the system based on the reward function introduced in this paper and the reward function presented in [3] and [27]. The simulations were run while keeping all the parameters same except for the reward function in the two instances.



(a) Proposed Method (b) [28] Based Method (c) Greedy Method (d) Random Method

FIGURE 6. Paths obtained by different methods for a UAV-BS. The blue dots indicate ground users (100 ground users distributed uniformly and at random in an area of $2 \times 2\,km^2$). The red dots indicate hovering points of the UAV-BS. The green circles indicate coverage areas. The numbers next to hovering points indicate time iterations of hovering at that point. It can be seen that the UAV-BS has been hovering for more than one iteration at some points.

- Total coverage of the ground users.
- Number of individual ground users covered.
- Fairness in providing coverage.

We compared the performance of the method proposed in this paper with the performance of the state, action spaces and reward function proposed in [27], along with two commonly used baseline methods: Random and Greedy method.

1) GREEDY METHOD

Since the key objective is to maximise coverage, in the Greedy method, at each time interval the action a_t is selected in a way that maximises the instantaneous user coverage.

2) RANDOM METHOD

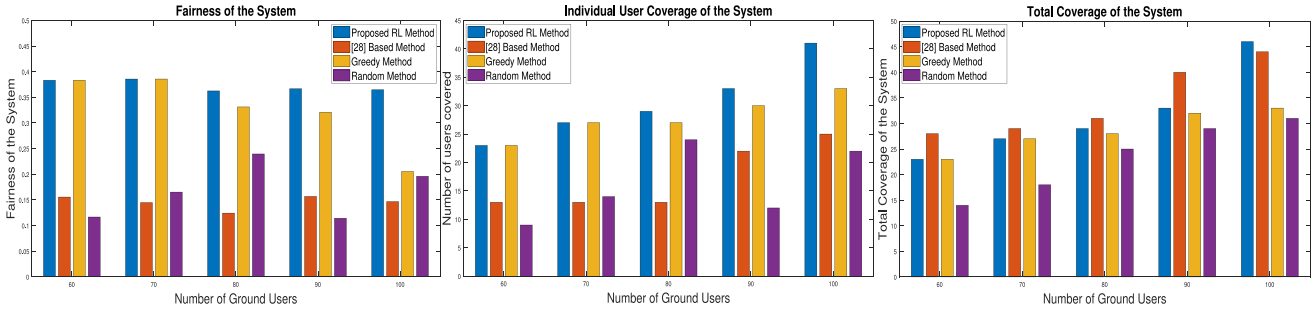
The UAV-BS would randomly select an action and perform the selected action. This process is repeated until the available energy is exhausted.

Since the method proposed in [27] assumes that the UAV-BS can cover multiple points of interest by hovering in one point, for simulating the [27] based method we discretise the area into smaller squares of width 250 m while keeping the UAV-BS coverage radius the same (Refer Fig. 6 for clarification).

B. SINGLE UAV-BS SCENARIO

First, we consider one UAV-BS providing coverage to the said geographical area. The effectiveness of the proposed method was tested with varying number of ground users and available energy units.

The first set of simulations were performed with the available energy set to 3000 units and with a varying number of ground users (60–100 users). The UAV-BS paths obtained by the proposed method and the three baseline methods considered are shown in Fig. 6. The proposed method prompts the UAV-BS to fly around the area serving a wide spread of users in comparison to the other methods, as seen in Fig. 6. Since the area covered by the proposed method is considerably higher, it can also be used for area screening and surveying purposes, in addition to providing coverage.



(a) System Fairness Comparison (b) Individual User Coverage Comparison (c) Total System Coverage Comparison

FIGURE 7. Comparison of system fairness, number of covered users and total coverage with a varying number of ground users. The results show that the proposed method results in better fairness and individual user coverage in comparison and the [27] based method results in better total system coverage in comparison.

The simulation results for the scenario of fixed manoeuvring energy with a varying number of ground users are give in Fig. 7.

The proposed RL method outperforms the baseline methods consistently in terms of the system fairness (Fig. 7a) and the number of individual users covered (Fig. 7b). The proposed RL method doubles the system fairness in comparison to the [27] based method and the Random method, system fairness is increased by an average of 20.1% in comparison to the Greedy method. The reward function of the proposed method prompts the UAV-BS to position itself in a manner that increases the system fairness, resulting in increased system fairness. However, according to (Fig. 7a), the system fairness decreases as the number of ground users increases. As the number of ground users increases, the number of hovering points to be covered in order to provide coverage to all ground users increases as well. However, the available energy of the UAV-BS remains the same. Therefore, this increases the number of uncovered users, resulting in low system fairness.

The proposed method increases the number of individual users covered by an average of 84.31% in comparison to the [27] based method and by an average of 8.16% compared to the Greedy method. The proposed method doubles the individual users covered in comparison to the Random method. As mentioned earlier, the proposed reward function prompts the system to increase system fairness in every iteration. According to (7), to increase fairness, number of individuals covered should be increased. Thus the proposed method strives to provide coverage to as many individual ground users as possible. As Fig. 7b shows, on average, the number of individual users covered has increased with the increase in the total number of ground users. As the total number of ground users increases in a fixed area, the user density increases. This increases the number of users that can be covered by hovering at a certain point, resulting in an increase of individual users covered.

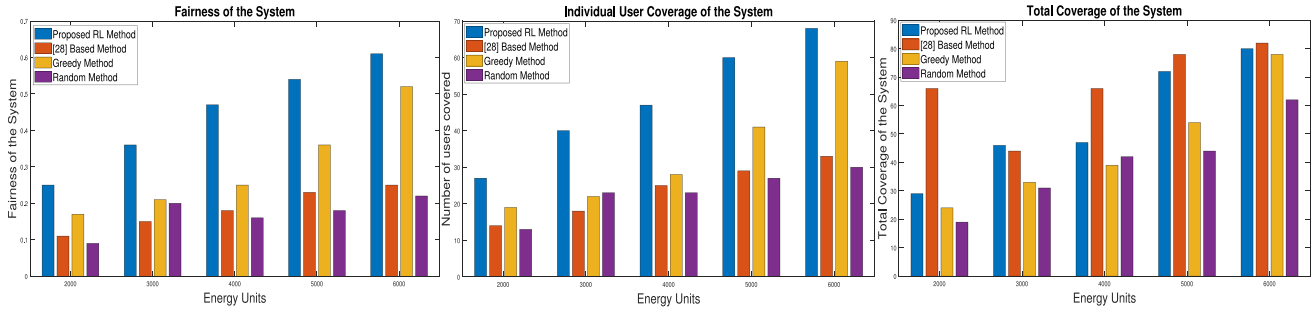
In terms of the total coverage of the system, the proposed method outperforms the Greedy and Random methods while [27] based method often shows better performance in comparison to all the other three methods. The proposed method has an average increase of 9.2% in total system coverage in comparison to Greedy method and 38.5% in comparison to Random method. The [27] based method has an average total system coverage increase of 8% in comparison to the proposed method. As discussed in the comparison of the reward functions in the proposed method and [27], the reward function in [27], makes the UAV-BS slightly hesitant to fly but prompts to increase system coverage. Since the total system coverage can be increased by providing constant coverage to the same subset of users (according to 6), the [27] based method shows better performance in total system coverage.

According to the results section in [27], the method proposed in [27] outperforms the Greedy method. However, in our simulations [27] inspired method was constantly outperformed by the Greedy method. It should be noted that the variant of Greedy method used in [27], tries to maximise the instantaneous reward while in our simulations the Greedy method tries to maximise the instantaneous user coverage.

In the second set of simulations, we keep the number of users constant (100 users) in the considered area and change the available energy in the UAV-BS (2000–6000 units) in order to understand the impact of available energy in the performance of the proposed method. The results of this set of simulations are shown in Fig. 8.

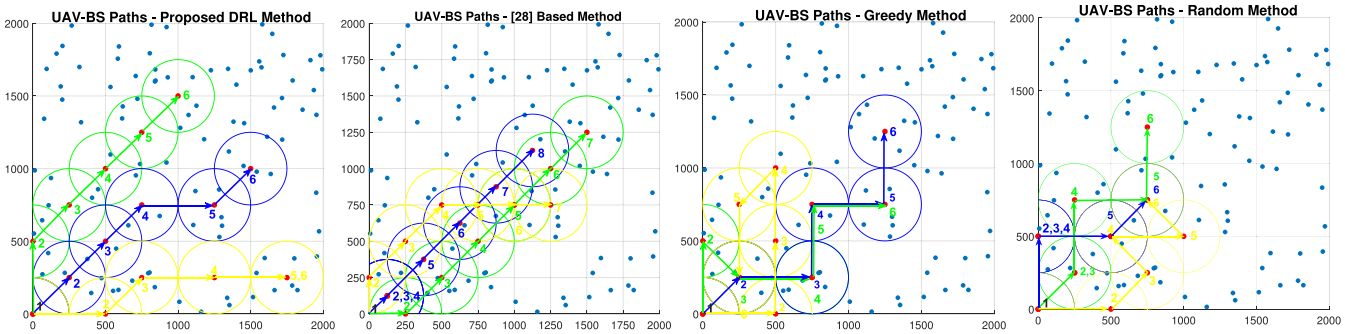
The fairness of the system increases with available energy in all three methods. As the available energy increases the UAV-BS's ability to fly increases, making it possible to provide coverage to a higher number of users. This increases fairness as well as individual and total coverage of the system. However, the proposed RL method outperforms the baseline methods in terms of the system fairness, the number of individual users covered while [27] based method has better total coverage.

The proposed RL method increases the system fairness by an average of 54.76% in comparison to the Greedy method. In comparison to the [27] based method and Random method, the proposed method doubles the system fairness on average. The number of individual users is increased by an average of 100% by the proposed method in comparison to the [27] based method and Random method. The increase in the individual



(a) System Fairness Comparison (b) Individual User Coverage Comparison (c) Total System Coverage Comparison

FIGURE 8. **Comparison of system fairness, number of covered users and total coverage with varying energy available for manoeuvring. The results show that the proposed method results in better fairness and individual user coverage in comparison and the [27] based method results in better total system coverage in comparison.**



(a) Proposed Method (b) [28] Based Method (c) Greedy Method (d) Random Method

FIGURE 9. **Paths obtained by different methods for a scenario of 3 UAV-BSs providing coverage. The blue dots indicate ground users (100 ground users distributed uniformly and at random in an area of $2 \times 2\,km^2$). The red dots indicate hovering points of the UAV-BS. The coverage areas of different UAV-BSs are shown in different colours. The time iterations of the UAV-BSs hovering at different points are shown by the numbers in different colours. All three UAV-BSs have the same initial point marked '1'.**

user coverage in comparison to the Greedy method is 50.68%. The proposed method outperforms the Greedy method by an average of 23.33% and the Random method by an average of 41.12% in terms of the total system coverage. The [27] based method outperforms the other considered methods in total system coverage, with an average increase of 18.1% in comparison to the proposed method. The reason for the increased total system coverage in the [27] based method was explained before.

Based on the above simulation results, we can conclude that the proposed RL based method outperforms all three baseline methods in system fairness and the number of individual users covered. The proposed RL based method outperforms Greedy method and Random method in terms of total system coverage as well. However, the [27] based method displays better performance in terms of total system coverage.

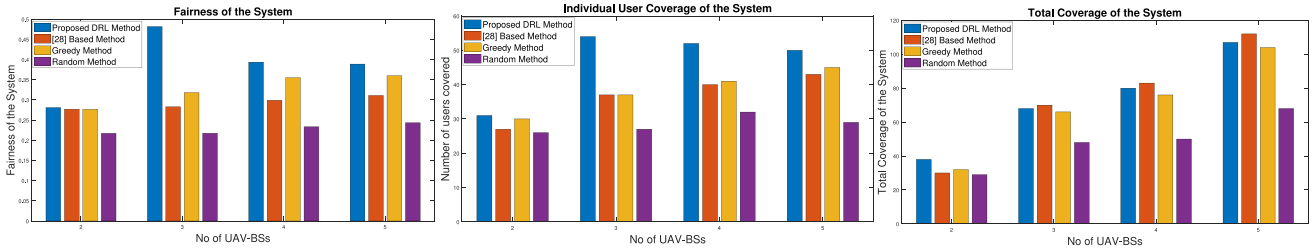
C. MULTI UAV-BS SCENARIO

In the multi-UAV scenario, we consider the same area size and coverage range for the simulations. The available energy for the UAV-BSs for flying is set to 2000 units. The number of ground users is fixed to be 100. The users are uniformly and at random distributed in the considered area. We increase the number of UAV-BSs one at a time (2–5) to observe its effect on the ground user coverage. The initial location of the UAVs is the lower left-most point of the area. Up to three UAV-BSs are dispatched from the same initial location. The additional UAV-BSs are dispatched from the next corner of the considered area anticlockwise.

Fig. 9 shows the UAV-BS paths given in the four methods–proposed DRL method, [27] based method, Greedy method and Random method, when 3 UAV-BSs are used to provide coverage to an area of 2×2 km with 100 users.

The figures clearly show that the paths given by the proposed method have no collisions and very low interference to ground users, in comparison to the paths given by the three baseline methods. It is an added advantage that the area covered by the UAV-BSs in the proposed method is higher than the baseline methods. Thus, an extension of the proposed method could be surveying the area in addition to providing coverage to the ground users.

The performance comparison between [27] based method, Greedy method, Random method, and the proposed DQL method can be seen in Fig. 10.



(a) System Fairness Comparison (b) Individual User Coverage Comparison (c) Total System Coverage Comparison

FIGURE 10. Comparison of system fairness, number of covered users and total coverage with varying number of UAV-BSs used to provide coverage. The results show that the proposed method results in better fairness and individual user coverage in comparison to all three baseline methods and [27] based method shows better performance in total system coverage.

The simulation results show that the proposed DRL method outperforms the baseline methods by a considerable margin with respect to system fairness (Fig. 10a). The proposed DRL method has increased the system fairness by an average of 32.08% in comparison to the [27] based method, by an average of 17.94% in comparison to Greedy method and 69.81% in comparison to Random method.

However, it can be noted that the system fairness in the proposed DRL method increases up to a threshold (3 UAV-BSs) and starts to decrease afterwards, while the fairness given by the baseline methods continues to increase with the number of UAV-BSs used. This is due to the increased possibility of inter-UAV collisions and interference to ground users with the increase of the UAV-BSs used. The proposed method aims to eliminate inter-UAV collisions and minimise interference to ground users. This reduces the UAV-BSs' freedom of flying. This results in UAV-BSs hovering in the same position for longer periods, resulting in reduced system fairness. However, the baseline methods do not take inter-UAV collisions and interference to ground users into consideration, thus, the UAV-BSs are not restricted of the freedom to fly. Hence, the number of users that can be covered increases with the number of UAV-BSs, which in return increases the system fairness. However, this increase of fairness comes at the cost of inter-UAV collisions and interference to ground users.

The above observation suggests that there exists an optimal number of UAV-BSs to be deployed to provide fair coverage to ground users, while avoiding inter-UAV collisions and minimising interference to ground users. We intend to analytically derive the optimal number of UAV-BSs to serve a specific area in our future work.

Fig. 10b shows the comparison to the number of individual users covered. The proposed DRL method shows an increased number of ground users covered in comparison to all three baseline methods. The proposed DRL method increases the individual user coverage by an average of 26.76% in comparison to the [27] based method, 21.8% in comparison to the Greedy method and 63.54% comparison to the Random method.

The simulation results show that with the proposed DRL method the number of individual users covered initially increase and start to decrease after a threshold (3 UAV-BSs in the considered scenario). The reason for this observation is explained earlier with respect to the system fairness.

The comparison to total system coverage is shown in Fig. 10c. The proposed DRL method out-performs the Greedy method and Random method in terms of total system coverage. However, the [27] based method shows better performance in total system coverage. The proposed method has increased the total system coverage by an average of 7.48% in comparison to the Greedy method and 47.51% in comparison to the Random method. The [27] based method shows an average increase of 3.65% in total system coverage in comparison to the proposed method. The reason for the [27] based method's increased total system coverage is explained earlier in this paper in the discussion of simulation results for the single UAV-BS scenario.

In the proposed DRL method, the total system coverage continues to increase with the number of UAV-BSs, in contrast with the system fairness and the number of individual users covered, which start to decrease after a threshold. This is because, the total system coverage can be increased even when the UAV-BSs have restricted freedom for flying. According to (6), the system coverage can be increased even when UAV-BSs continue to hover in the same position providing coverage to the same subset of users. Thus, the reduced freedom in flying with the increasing number of UAV-BSs does not affect the performance in terms of the total system coverage. Thus, since the number of users that can be covered in an average time instance increases with the number of UAV-BSs, the total system coverage increases with the number of UAV-BSs.

One key objective of the proposed method is to reduce the interference to the ground users. Fig. 11 shows that the instances of interference occurrences are considerably low in the proposed approached in comparison to the three baseline methods. The average reduction of occurrences of interference to ground users by neighbouring UAV-BSs in comparison to the [27] based method is 26.15%, in comparison to the Greedy method is 28.13% and in comparison to the Random method is 41.43%.

D. CONVERGENCE OF THE PROPOSED DQL ALGORITHM

In the proposed DQL algorithm we leverage penalties for UAV-BSs flying over the boundaries of the considered region,

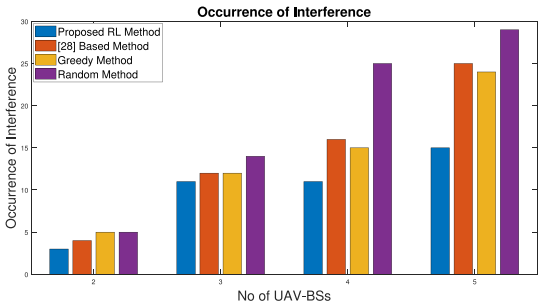


FIGURE 11. Occurrences of Interference.

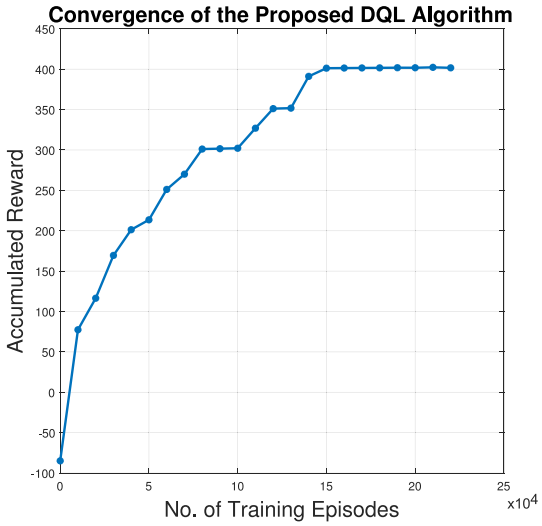


FIGURE 12. Convergence Performance of the Proposed DQL Algorithm.

interference to ground users, and collisions between UAV-BSs. Multiple penalties might lead to divergence in DQL algorithms. However, extensive simulations show the parameters we have applied for the penalties result in convergence of the algorithm within a reasonable number of training episodes. The convergence of the proposed DQL algorithm is shown in Fig. 12. The average accumulated reward is shown in Fig. 12, in a scenario of two UAV-BSs with 100 ground users. The convergence plot shows that the accumulated reward of the system remains low in the early episodes, however after a considerable number of training episodes, the accumulated reward reaches a maximum and remains stable.

VII. CONCLUSION

UAV-BSs are an effective method of providing wireless coverage ground users. However, when deploying UAV-BSs several factors need to be considered - energy limitations, collisions between UAV-BSs, interference to ground users, and fairness of the system. Due to the complexity and interdependencies of these factors, optimal path finding problem for UAV-BSs is too challenging to be solved by conventional optimisation problem solving methods. In this paper, we propose an RL based method for the scenario of using one UAV-BS and a DRL based method for using a fleet of UAV-BSs.

Simulation results show that the proposed methods outperform the baseline techniques in terms of the total coverage of the system by an average increase of 15.04%, the number of individual users covered by an average increase of 42.54%, system fairness by an average increase of 38.94%. Further, the proposed DQL method reduces the interference to ground users and inter-UAV collisions in comparison to the baseline methods.

REFERENCES

[1] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, "Placement optimization of UAV-mounted mobile base stations," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 604–607, Mar. 2017.

[2] A. Fotouhi, M. Ding, and M. Hassan, "Dronecells: Improving 5G spectral efficiency using drone-mounted flying base stations," *J. Trans. Mobile Comput.*, Jul. 2017. [Online]. Available: https://arxiv.org/abs/1707.02041, Accessed on: Dec. 15, 2019.

[3] C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, "Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2059–2070, Sep. 2018.

[4] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, and E. Dutkiewicz, "Algorithm for energy efficient inter-UAV collision avoidance," in *Proc. 17th Int. Symp. Commun. Inf. Technol.*, Sep. 2017, pp. 1–5.

[5] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Commun. Surveys Tutorials*, vol. 18, no. 4, pp. 2624–2661, Oct.–Dec. 2016.

[6] J. Li *et al.*, "Joint optimization on trajectory, altitude, velocity and link scheduling for minimum mission time in UAV-aided data collection," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1464–1475, Feb. 2020.

[7] H. Mei, K. Yang, Q. Liu, and K. Wang, "Joint trajectory-resource optimization in UAV-enabled edge-cloud system with virtualized mobile clone," *IEEE Internet Things J.*, to be published.

[8] S. Zhang, H. Zhang, Q. He, K. Bian, and L. Song, "Joint trajectory and power optimization for UAV relay networks," *IEEE Commun. Lett.*, vol. 22, no. 1, pp. 161–164, Jan. 2018.

[9] X. Zhu, Y. Liang, and M. Yan, "A flexible collision avoidance strategy for the formation of multiple unmanned aerial vehicles," *IEEE Access*, vol. 7, pp. 140 743–140 754, 2019.

[10] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, and E. Dutkiewicz, "Potential field based inter-UAV collision avoidance using virtual target relocation," in *Proc. IEEE 87th Veh. Technol. Conf.*, Jun. 2018, pp. 1–5.

[11] S. H. Arul *et al.*, "LSwarm: Efficient collision avoidance for large swarms with coverage constraints in complex urban scenes," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3940–3947, Oct. 2019.

[12] 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.

[13] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, and E. Dutkiewicz, "Empirical power consumption model for UAVs," in *Proc. IEEE 88th Veh. Technol. Conf.*, Aug. 2018, pp. 1–5.

[14] S. Kandeepan, K. Gomez, L. Reynaud, and T. Rasheed, "Aerial-terrestrial communications: Terrestrial cooperation and energy-efficient transmissions to aerial base stations," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2715–2735, Oct. 2014.

[15] 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.

[16] C. Lai, C. Chen, and L. Wang, "On-demand density-aware UAV base station 3D placement for arbitrarily distributed users with guaranteed data rates," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 913–916, Jun. 2019.

[17] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage," *IEEE Wireless Commun. Lett.*, vol. 6, no. 4, pp. 434–437, Aug. 2017.

[18] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 38–41, Feb. 2018.

[19] L. Liu, S. Zhang, and R. Zhang, "Comp in the sky: UAV placement and movement optimization for multi-user communications," *IEEE Trans. Commun.*, vol. 67, no. 8, pp. 5645–5658, Aug. 2019.
[20] H. V. Abeywickrama, Y. He, E. Dutkiewicz, and B. A. Jayawickrama, "An adaptive UAV network for increased user coverage and spectral efficiency," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2019, pp. 1–6.
[21] A. Fotouhi, M. Ding, and M. Hassan, "Flying drone base stations for macro hotspots," *IEEE Access*, vol. 6, pp. 19 530–19 539, Mar. 2018.
[22] X. Zhang and L. Duan, "Fast deployment of UAV networks for optimal wireless coverage," *IEEE Trans. Mobile Comput.*, vol. 18, no. 3, pp. 588–601, Mar. 2019.
[23] A. V. Savkin and H. Huang, "Deployment of unmanned aerial vehicle base stations for optimal quality of coverage," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 321–324, Feb. 2019.
[24] U. Challita, W. Saad, and C. Bettstetter, "Cellular-connected UAVs over 5G: Deep reinforcement learning for interference management," *IEEE Trans. Wireless Commun.*, to be published.
[25] N. Cheng *et al.*, "Space/aerial-assisted computing offloading for IoT applications: A learning-based approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, May 2019.
[26] L. Xiao *et al.*, "Reinforcement learning based downlink interference control for ultra-dense small cells," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 423–434, Jan. 2020.
[27] C. H. Liu, X. Ma, X. Gao, and J. Tang, "Distributed energy-efficient multi-UAV navigation for long-term communication coverage by deep reinforcement learning," *IEEE Trans. Mobile Comput.*, to be published.
[28] X. Liu, Y. Liu, and Y. Chen, "Reinforcement learning in multiple-UAV networks: Deployment and movement design," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8036–8049, Aug. 2019.
[29] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, and E. Dutkiewicz, "Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance," *IEEE Access*, vol. 6, pp. 58 383–58 394, 2018.
[30] M. Ding and D. Lopez Perez, "Please lower small cell antenna heights in 5G," in *Proc. IEEE Global Commun. Conf.*, Dec. 2016, pp. 1–6.
[31] U. Mengali, *Synchronization Techniques for Digital Receivers*. Berlin, Germany: Springer, 2013.
[32] U. Challita, W. Saad, and C. Bettstetter, "Deep reinforcement learning for interference-aware path planning of cellular-connected UAVs," in *Proc. IEEE Int. Conf. Commun.*, May 2018, pp. 1–7.
[33] "Mavlink developer guide," [Online]. Available: https://mavlink.io/en/. Accessed on: Jan. 5, 2020.
[34] "Qgroundcontrol home," 2019. [Online]. Available: http://qgroundcontrol.com/, Accessed on: Jan. 5, 2020.
[35] H. Shi, R. V. Prasad, E. Onur, and I. G. M. M. Niemegeers, "Fairness in wireless networks: Issues, measures and challenges," *IEEE Commun. Surv. Tut.*, vol. 16, no. 1, pp. 5–24, Mar. 2014.
[36] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination," Eastern Research Laboratory, Digital Equipment Corporation, Hudson, MA, 1984.
[37] G. Laporte, "The traveling salesman problem: An overview of exact and approximate algorithms," *Eur. J. Oper. Res.*, vol. 59, pp. 231–247, 1992.
[38] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 279–292, May 1992. [Online]. Available: https://doi.org/10.1007/BF00992698
[39] G. Laporte, "Deep reinforcement learning: An overview," *Cornell University*, Nov. 2018. [Online]. Available: https://arxiv.org/abs/1701.07274v6. Accessed on: Dec. 15, 2019.