



Article

Reinforcement Learning Aided UAV Base Station Location Optimization for Rate Maximization

Sudheesh Puthenveetil Gopi ^{1,*}  and Maurizio Magarini ² 

¹ Department of E&C, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India

² Politecnico di Milano, 20133 Milan, Italy; maurizio.magarini@polimi.it

* Correspondence: sudheesh.pg@manipal.edu

Abstract: The application of unmanned aerial vehicles (UAV) as base station (BS) is gaining popularity. In this paper, we consider maximization of the overall data rate by intelligent deployment of UAV BS in the downlink of a cellular system. We investigate a reinforcement learning (RL)-aided approach to optimize the position of flying BSs mounted on board UAVs to support a macro BS (MBS). We propose an algorithm to avoid collision between multiple UAVs undergoing exploratory movements and to restrict UAV BSs movement within a predefined area. Q-learning technique is used to optimize UAV BS position, where the reward is equal to sum of user equipment (UE) data rates. We consider a framework where the UAV BSs carry out exploratory movements in the beginning and exploitative movements in later stages to maximize the overall data rate. Our results show that a cellular system with three UAV BSs and one MBS serving 72 UE reaches 69.2% of the best possible data rate, which is identified by brute force search. Finally, the RL algorithm is compared with a K-means algorithm to study the need of accurate UE locations. Our results show that the RL algorithm outperforms the K-means clustering algorithm when the measure of imperfection is higher. The proposed algorithm can be made use of by a practical MBS–UAV BSs–UEs system to provide protection to UAV BSs while maximizing data rate.

Keywords: UAV BS; reinforcement learning; K-means clustering



Citation: Gopi, S.P.; Magarini, M. Reinforcement Learning Aided UAV Base Station Location Optimization for Rate Maximization. *Electronics* **2021**, *10*, 2953. <https://doi.org/10.3390/electronics10232953>

Academic Editor: Nurul I. Sarkar

Received: 27 October 2021

Accepted: 25 November 2021

Published: 27 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The use of unmanned aerial vehicles (UAVs) in combination with terrestrial communication networks, in various capabilities, was initially considered for long-term evolution (LTE). In LTE, UAVs were considered both as flying user equipment (UE), widely known as cellular connected UAVs, and as flying base station (BS). The cellular connected UAVs were used extensively for surveying, acquiring sensor data etc., whereas the UAV BSs are proposed to play a major role especially during natural calamities and similar situations where ground based structure might be absent [1]. Apart from providing the necessary communication service during a natural calamity, a UAV can also be used to improve the performance of an existing network. However, the UAV BSs are still costly, and researchers from all over the world are working on prototypes of various capacities. In 2020, Verizon experimented with a 300 pound prototype [2], clearly concluding that a cost effective UAV BS is not yet available.

Despite the contribution of UAVs in cellular communication, the UAV requires considerable overhead and additional training to provide satisfactory performance, and machine learning (ML) is used extensively to achieve the same [3]. ML algorithms in different categories are used extensively in optimization and automation purposes. Optimizing multiple network parameters, intelligently adapting to modulation schemes, and optimization in networked UAVs are just few among many research activities. Recent research considers positioning of UAVs using ML algorithms to improve spectral efficiency, increase

coverage, and/or provide connectivity during disasters [4]. Various categories of algorithms in supervised, unsupervised, and reinforcement learning (RL) division are used in the literature.

One of the main challenges in using supervised and unsupervised algorithms is to have a huge set of training data, whereas RL does not require any training. RL, on the other hand, learns from the environment (observes different parameters such as data rate) and improves the performance with time. However, the complete possibilities of RL, considering practical deployment, in intelligent positioning of UAV BS along with macro base station (MBS), have not been extensively studied. In this paper, we consider maximizing the overall data rate by intelligent deployment of UAV BS in the downlink of a cellular system. We investigate an RL-aided approach to optimize the position of UAV BSs to support the MBS.

The main contribution of this paper is the application of RL to optimize UAV BSs position in UAV BSs–MBS cellular architecture, with an eye to practical deployment. In addition to application of RL, considering the practical UAV BSs deployment, we also use: (i) an algorithm to avoid collision between multiple UAV BSs; (ii) a greedy approach to enhance the learning process in initial stages and achieve for data rate in later stages.

The paper is organized as follows. A review of related works is given in Section 2. Section 3 provides system model and the problem formulation. In Section 4, we provide a basic introduction to RL and further elaborate the application of RL to UAV BS position optimization. Simulation results are presented in Section 5, comparisons with other works are mentioned in Section 6, and finally, conclusions are drawn in Section 7.

2. Related Work

Various ML algorithms are used extensively in UAV-related research [5–15]. Energy spent on UAV movement is a bottleneck in determining the operating time of the UAV. Optimizing energy in UAV by trajectory optimization is discussed in [5]. Another work focuses on resource allocation in a cache-enabled UAV network [6]. A solution for interference management using artificial neural networks (ANN) is proposed for UAV networks in [7]. UAVs are also used in image processing and object detection, and they find application in agriculture and forest based research [8].

UAV BSs are also suggested to provide swift connectivity in disaster resilient networks [9]. In fact, under such circumstances, they can replace a fixed pico BSs (PBSs), thus resulting in a UAV PBS assisted heterogeneous network (HetNet) architecture [10]. The deployment of UAV BSs in HetNet with a central unit minimizes the inter cell interference and leads to a cell-free network [11].

Considering the randomness in the UEs positions, predetermined movements of UAV BS will not bring the advantages as envisaged. A sectoring-based approach is proposed to provide 2D placement of UAV BS in [16]. Various UAV BS positioning strategies were proposed, where the problem of UAV BS positioning is framed as an optimization problem [17–20]. A UAV BS can use artificial intelligence (AI) to move to the best position, thereby enhancing the performance [3,21]. ML, which is a branch of AI, provides the ability to learn and adapt to situations through experiences. UAV BSs deployed with the help of ML algorithms can provide a reliable service to users, despite the user pattern variation. Different strategies of ML, such as bio-inspired algorithms, unsupervised, and reinforcement learning (RL), have already been considered for optimal positioning of UAV BS in [22,23]. In [22], authors implemented a three dimensional UAV positioning with K-means clustering algorithm, an unsupervised ML algorithm that groups UEs to neighbouring cluster heads. Another strategy is to use device-to-device communication to expand the coverage of the UAV BS using clustering algorithms [23].

In this paper, we use RL technique to learn and adapt based on the responses from the environment. Approaches with RL, where UAVs are used in HetNet, are considered in [24–26]. The work in [24] considers UAV BSs along with terrestrial networks. However, in [24], multiple MBSs along with single UAV BS are considered, which is not the case in a practical HetNet architecture. The optimization of the UAV BSs position in the downlink

of a UAV BSs-UE network for overall data rate maximization is considered in [26]. An extended version of [26], where an optimized UAV positioning algorithm with an MBS that provides backhaul to UAV BS is considered in [25]. The UAV BS acts as relay, and the capacity of the system is defined by the minimum of backhaul and UAV-BS to UE communication in [25]. However, we consider both MBS and UAV BS to serve users, unlike in [25]. Table 1 lists related works and comparisons with existing works.

Table 1. Comparison with the existing work.

Ref.	Highlighted	Technique Used	Limitation and Future Directions
[16]	2D UAV BS positioning	-Sectoring-based UAV positioning	not an ML-based solution.
[17]	Efficient placement of UAV BS to maximize revenue	-Mixed integer non-linear optimization problem -Focus on revenue maximization	not an ML-based solution
[18]	Placement Optimization of UAV MBS	-Polynomial time algorithm	non-ML algorithm, which cannot be used with dynamic UE patterns
[19]	UAV BS position optimization using non ML optimization	-Brute force search -Maximal weight area algorithm to maximize coverage	non-ML solution
[20]	UAV trajectory optimization	-Framed as a convex optimization problem	-Applicable to fixed wing UAV at fixed height
[22]	Data-driven UAV position optimization	-K-means clustering algorithm -Fixed user positions obtained using PPP	UE locations must be known prior to optimization
[24]	UAV BS position optimization using Q-learning	-RL based	Single UAV BS is used
[25,26]	Same as above	-Rate maximization -Q-learning	-MBS is not used to serve UEs -Concept of varying exploration and exploitation is not attempted -Practical aspects like UAV BS collision etc are not addressed
[27]	UAV BS trajectory design using RL	-Deterministic Policy Gradient algorithm -Applied in uplink	-Computationally complex
[28]	Q-learning-assisted UAV trajectory design	-UAV BS changes its position based on UE positions	Scenario considered is unreal and limited to 15×15 square

The main contribution of our work compared with existing work is as follows: (i) the MBS is considered only as backhaul in [25], whereas, in this work, the MBS serves users, in addition to backhaul; (ii) we make use of a scheme proposed in [29], where the UAV BSs explore more in the initial stages and exploit in later stages, with an aim to maximize the overall data rate; (iii) in addition to implementation of RL assisted UAV BS positioning, we also propose a scheme to avoid collision among multiple UAV BSs; and (iv) considering the exploratory nature of UAV BS, we propose a method to avoid UAV BSs from moving out of the desired service area.

3. System Model

We consider a UAV-assisted cellular architecture with one MBS and N UAV BSs. Let us define the set $\mathbf{BS} = \{BS_0, BS_1, \dots, BS_N\}$, where BS_0 corresponds to the MBS and $\{BS_1, \dots, BS_N\}$ represent N UAV BSs. The UAV-assisted cellular system serves M users, denoted by the set $\mathbf{UE} = \{UE_1, UE_2, \dots, UE_M\}$. The users are distributed using poisson point process (PPP). We assume that a UE is connected to one BS at a time.

Figure 1 shows the system model with an MBS, N UAV BSs, and M UEs. We consider a downlink system having M sub-channels, each having bandwidth W , thereby making the total bandwidth allotted to the system $M \times W$. We also assume that each UAV BS has the same transmitting power P_{UAV} , while that of the MBS is P_{MBS} . As in [30], we use partially shared deployment (PSD) strategy to allocate spectrum to MBS and UAV BSs. That is, out of K_{tot} sub-channels allocated to MBS-UAV BSs, K_{sh} channels are shared by UAV BSs and MBS, while the remaining ($K_{ex} = K_{tot} - K_{sh}$) ones are specific to the MBS. This resource allocation strategy is appropriate for a cell-free architecture, where a user obtains

the same spectrum even after changing the position. The notations and their descriptions are summarized in Table 2.

Table 2. Notations and description.

Variable	Description
N, M	Number of UAV BS and UE respectively
W	Bandwidth
P_{UAV}, P_{MBS}	Transmit power of UAV BS and MBS respectively
K_{tot}	Total sub-channels allocated to MBS-UAV BSs
K_{sh}	Number of channels shared by UAV BSs and MBS
K_{ex}	Number of channels specific to MBS
d_{ij}	Distance between BS j and UE i
$(x_{BS}^j, y_{BS}^j, h_{BS}^j)$	Coordinates of BS j
$(x_{UE}^i, y_{UE}^i, 0)$	Coordinates of UE i
$PL(d_{ij}), PL(d_{i0})$	PL associated with BS j and UE i , PL associated with MBS and UE i
G_{ij}, G_{i0}	Channel gain between BS j and UE i , channel gain between MBS and UE i
$SINR_{ij}^k, SINR_{i0}^k$	SINR at UE i using the sub-channel k of BS j , SINR at UE i using the sub-channel k of MBS
$SINR_{ij}^l$	SINR at UE i using exclusive sub-channel l of MBS
R_{ij}, R_{ij}	Data rate of UE i associated to BS j , data rate of UE i associated to MBS
α_{ij}	UE i 's normalized time of association with BS j
γ	SINR threshold
r, s, a	Reward, State, Action
s', a'	Next state, set of actions that can be performed when agent is in s'
β	Discount factor
ϵ	Parameter defining probability of exploration-exploitation movements
ψ	Decay rate
J	Loss function
ψ	Decay rate
π	Parameter specifying association of point to cluster
μ	Vector carrying location of centroid
ρ	Parameter specifying the effect of imperfection in position data

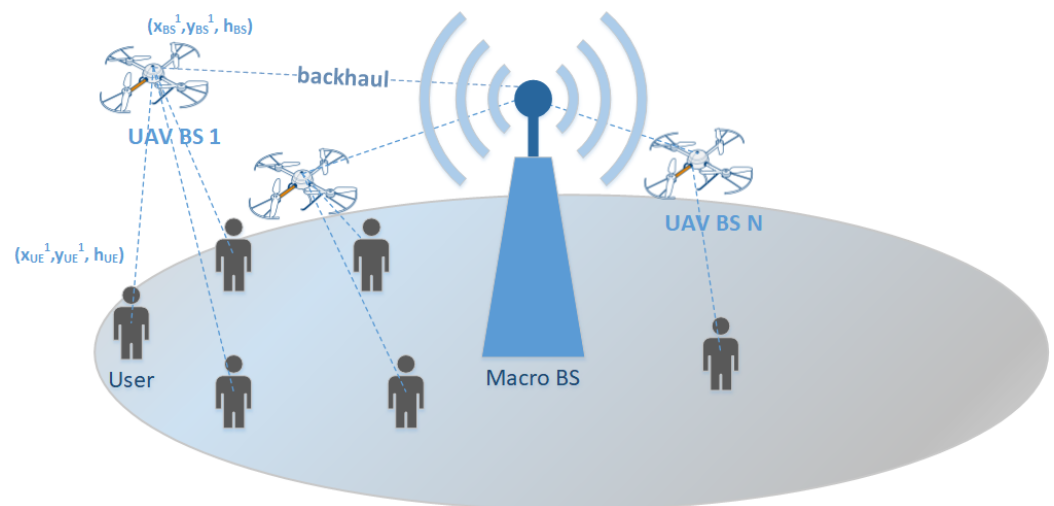


Figure 1. Architecture of the UEs–UAV BSs–MBS system.

The distance between BS $j, j \in \{0, 1, \dots, N\}$, and UE $i, i \in \{1, \dots, M\}$, can be expressed as

$$d_{ij} = \sqrt{(x_{UE}^i - x_{BS}^j)^2 + (y_{UE}^i - y_{BS}^j)^2 + (h_{BS}^j)^2}, \quad (1)$$

where $(x_{BS}^j, y_{BS}^j, h_{BS}^j)$ and $(x_{UE}^i, y_{UE}^i, 0)$ represent the coordinates of BS j and UE i , respectively. The path loss (PL) associated with MBS and UE i can be expressed as [30]

$$PL(d_{i0})[\text{dB}] = 128.1 + 37.6 \log_{10}(d_{i0}/1000) [\text{dB}]. \quad (2)$$

Similarly, the PL associated with UAV BS j and UE i can be expressed as [31]

$$PL(d_{ij})[\text{dB}] = 92.45 + 20 \log_{10}(d_{ij}/1000) [\text{dB}]. \quad (3)$$

Subsequently, we can write channel gain $G_{ij}[\text{dB}]$, for a given distance d_{ij} , as

$$G_{ij}[\text{dB}] = PL(d_{ij})[\text{dB}] + \zeta[\text{dB}] - \xi[\text{dB}], \quad (4)$$

where $\zeta[\text{dB}]$ and $\xi[\text{dB}]$ represent shadow fading and antenna gain, respectively.

The signal-to-interference noise ratio (SINR) at UE i using the shared sub-channel k of UAV BS j can be expressed as

$$\text{SINR}_{ij}^k = \frac{P_u G_{ij}}{N_0 + P_u \sum_{h=0, h \neq j}^N G_{ih}}, \quad (5)$$

where $P_u = P_{UAV}/K_{sh}$, and N_0 is the noise power. Without loss of generality, we can calculate the SINR at UE j using shared sub-channel k of MBS as

$$\text{SINR}_{i0}^k = \frac{P_u G_{i0}}{N_0 + P_u \sum_{h=1}^N G_{ih}}. \quad (6)$$

Unlike sub-channels shared by UAV BSs and MBS, the use of an exclusive sub-channel in MBS does not result in interference. Thus, SINR at UE i using exclusive sub-channel l of MBS can be obtained from

$$\text{SINR}_{i0}^l = \frac{P_m G_{i0}}{N_0}, \quad (7)$$

where $P_m = \frac{P_{MBS} - P_{UAV}}{K_{ex} - K_{sh}}$ and $P_{MBS} > P_{UAV}$. The data rate of the i th UE associated with MBS is written as

$$R_{i0} = K_{ex} W \log_{10}(1 + \text{SINR}_{i0}^l) + K_{sh} W \log_{10}(1 + \text{SINR}_{i0}^k). \quad (8)$$

Similarly, the data rate of UE i associated to UAV BS j can be expressed as

$$R_{ij} = K_{sh} W \log_{10}(1 + \text{SINR}_{ij}^k). \quad (9)$$

Problem Formulation

The data rate experienced by UE i can be written as

$$R_i = \sum_{j=0}^N \alpha_{ij} R_{ij}, \quad (10)$$

where α_{ij} represents the scheduling factor, which describes UE i 's normalized time of association with BS j [30]. The UE is connected to the UAV BS, for which there is the maximum SINR. Additionally, we associate a UE with the MBS, if the maximum of SINR offered by UAV BS is less than a predefined threshold γ . That is, if $\max(\text{SINR}_{mn}) < \gamma$, the UE m gets connected to MBS.

Considering that UAV BSs vary the position, the problem can be formulated as the maximization of the sum-rate of the entire UE-UAV BS-MBS system as

$$\arg \max_{(x_i, y_i)} \sum_{i=1}^M R_i / M. \quad (11)$$

Assuming a UAV BS at fixed height, the problem can be formulated as an optimal two-dimensional UAV BS positioning one. We rely on RL to find the optimal position of UAV BS.

4. Reinforcement Learning

RL is a branch of ML, where the agent learns and adapts to the situations based on the environment it is in. The agent performs action a from state s and reaches state s' by receiving a reward r . The agent performs a fixed number of steps called 'episodes' and learns by collecting entries to the Q-table. A Q-table is a matrix, with dimension possible states \times possible actions, that finally makes the agent intelligent and allows it to choose the best action at a particular state. The agent can either choose an action that gives best reward (exploit) or can explore more options in order to maximize the cumulative discounted reward.

The Q-table is updated using the following equation:

$$Q(s, a) = r + \beta \max_{a'} Q(s', a'), \quad (12)$$

where the agent gets reward r on performing action a in state s and move to state s' , a' is the set of actions that can be performed when agent is in state s' , and $\beta \in [0, 1)$ is the discount factor. The Q value update is done by using the well-known Bellman equation

$$Q'(s, a) = Q(s, a) + \lambda [r + \beta \max_{a'} Q(s', a') - Q(s, a)], \quad (13)$$

where the value of $\lambda \in [0, 1)$ determines how quickly Q values change. We can see from (12) that $[r + \beta \max_{a'} Q(s', a')]$ is the target Q-value and $Q(s, a)$ gives the estimated Q-value.

4.1. Application of RL

In order to implement the Q-learning algorithm, it is mandatory to define the agent, the environment, and the associated states, actions, and rewards, as detailed in the following:

4.1.1. Environment

Set defined by M fixed UEs, represented by $\{UE_1, UE_2, \dots, UE_M\}$, and the MBS. Based on the movement of UAV BS, UEs connect and communicate via UAV BS and/or MBS.

4.1.2. Agent

The set of N UAV BSs acts as the agent, owing to the idea that the agent performs action a from state s and moves to next state s' . In this work, N UAV BSs can move by maintaining a fixed height. Subsequently, UEs are connected to any of UAV BS or to the MBS.

4.1.3. State

In practice, each UAV BS can move around a vast area, and each possible UAV BS position must be taken into account, while defining the states. This increases the number of states, resulting in delayed UAV BSs optimization. Therefore, to reduce computational complexity, we reduce the number of states by allowing UAV BS to move in incremental distances d_{sep} . The UAV BS can hold positions with a distance d_{sep} . We also restrict UAV BS movement to these grid points in the square grid. Consider a 2D area $(N_{grid} \times d_{sep}) \times (N_{grid} \times d_{sep})$, where N_{grid} is the number of grid points in a row or column, and the number of grid points turn out to be $N_{grid} \times N_{grid}$. Given the area, reducing d_{sep} or increasing grid points increases the number of states and hence the time required for computation. However, it is not necessary to restrict the number of states if the server processing RL algorithm can handle the computational load.

4.1.4. Action

The agent can choose an action a from set of actions $\mathcal{A}(s)$. We assume that each UAV BS can either move east/west, north/south, or maintain the same position. Therefore, five actions are possible for each UAV BS, and the combined number of actions is given by 5^N .

4.1.5. Reward

We define reward as

$$r = \sum_{i=1}^M R_i / M. \quad (14)$$

It is worth noting that reward is a parameter in measuring data rate resulting from UAV BS locations. Note that reward varies with UAV BS positions and associations.

4.1.6. Training Process

The agent's learning process starts from an initial state and carries out numerous transitions through different states and ends at a terminal state. After performing each action, the Q-table is updated. The Q-learning algorithm used in this paper is exemplified in Algorithm 1.

Algorithm 1: Proposed algorithm based on Q-learning

Input: UE number and locations, learning rate $\lambda \in (0, 1]$, epsilon (ϵ), maximum epsilon (ϵ_{max}), minimum epsilon (ϵ_{min}), decay rate (ψ).
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+$, $a \in \mathcal{A}(s)$, arbitrarily, except that $Q(\text{terminal}, \cdot) = 0$;
while $episode \neq 0$ **do**
 reset UAV BS locations;
 Initialize S ;
 while $step\ of\ episode \leq maximum\ number\ of\ episodes$ **do**
 Choose action a from s using policy derived from Q (ϵ -greedy);
 Take action a ; Reward, $r = \sum_{i=1}^M R_i / M$;
 observe next state s' ;
 $Q(s, a) \leftarrow Q(s, a) + \lambda[r + \beta \max_a Q(s', a) - Q(s, a)]$;
 $s \leftarrow s'$;
 $\epsilon = \epsilon_{min} + (\epsilon_{max} - \epsilon_{min}) \times \exp(-\psi \times episode)$
 while UAV BS location $(x_{BS}^j, y_{BS}^j, h_{BS}^j) \neq (0, 0, 0)$ **do**
 if $((x_{BS}^j > N_{grid} \times d_{sep}) \vee (y_{BS}^j > N_{grid} \times d_{sep}))$ **then**
 $r = -\infty$; /* To prevent movement of UAV BS outside
 required area */
 else
 if $(x_{BS}^j, y_{BS}^j, h_{BS}^j) \neq (x_{BS}^k, y_{BS}^k, h_{BS}^k)$, where $k \neq j$ **then**
 $r = -\infty$; /* To prevent UAV BS collision */
 else
 use Equation (14);
 end if
 end if
 end while
 end while
end while

To improve the overall data rate, we implement a framework where the UAV BSs carry out exploratory movements in the beginning and exploitative movements in later stages. We allow a UAV BS to explore newer locations by performing random actions at the beginning. With a lot of Q-table entries acquired using exploration, the agent performs exploitative actions in the subsequent episodes [29].

This is undertaken by varying ϵ according to the equation

$$\epsilon = \epsilon_{min} + (\epsilon_{max} - \epsilon_{min}) \times \exp(-\psi \times episode), \quad (15)$$

where ϵ , ϵ_{max} , ϵ_{min} , and ψ are epsilon value, maximum epsilon value, minimum epsilon value, and decay rate, respectively. When ϵ is close to zero, exploitative movement is more, and when ϵ is near one, exploration movement is more.

4.1.7. Collision Avoidance and Limiting UAV Location

In Algorithm 1, each episode begins from an initial state, where all UAV BSs have certain initial positions. Further, to prevent UAV BS collision, we check $(x_{BS}^j, y_{BS}^j, h_{BS}^j) \neq (x_{BS}^k, y_{BS}^k, h_{BS}^k)$, where $k \neq j$. That is, we prohibit UAV BS j and k to be at same location by setting reward $r = -\infty$. By setting the reward to $-\infty$, we prevent such future occurrences.

Similarly, we assign $r = -\infty$, when $((x_{BS}^j > N_{grid} \times d_{sep}) \vee (y_{BS}^j > N_{grid} \times d_{sep}))$ to prevent situations where UAV BSs explore outside specified area. We provide reward $r = -\infty$ for an action to move the UAV BS from edge of the specified area to a position outside the specified area. Algorithm 1 also describes the strategy used to avoid any UAV BS movement outside the grid locations. To analyze the performance of the proposed algorithm, we compare results of the proposed algorithm with K-means algorithm.

4.2. K-Means Clustering

K-means clustering is a very powerful algorithm that falls under the category of unsupervised ML algorithm. The role of the algorithm is to cluster data points into K non-overlapping subsets called clusters.

The K-means algorithm operates N data points, where each data point is represented by $x \in \{1, 2, \dots, N\}$. The goal is to find the association r , such that it minimizes the loss function,

$$J = \sum_n \sum_k \pi_{nk} \|x_n - \mu_k\|, \quad (16)$$

where

$$\pi_{nk} = \begin{cases} 1, & \text{if } k = \text{argmin}_j \|x_n - \mu_j\|. \\ 0, & \text{otherwise.} \end{cases} \quad (17)$$

This means that the value of π_{nk} is set to 1 if the data point x_i is assigned to cluster k and 0 for other clusters. μ_k is an n -dimensional vector that carries the location of centroid of the cluster.

4.3. Effect of Imperfect Locations in K-Means

Though K-means algorithm (Algorithm 2) provides optimal location of UAV BSs after multiple iterations, unlike RL, the algorithm requires knowledge of UEs location prior to optimization. If the locations given to K-means algorithm has imperfection, then the UAV BSs will be positioned in completely different locations. To understand this behaviour, we model the imperfection in position data as

$$\hat{x} = \rho x + (1 - \rho)\phi, \quad (18)$$

where x is the set of UE locations with size $M \times 2$, ρ is the effect of imperfection, where $0 < \rho < 1$ and ϕ is normally distributed matrix with size $M \times 2$.

Unlike K-means algorithm, RL does not need prior UE locations to compute UAV BSs locations. The RL algorithm takes r as the parameter to decide UAV BSs location, which is independent of location information x .

Algorithm 2: K-means algorithm based UAV BS positioning

Data: M UE positions where each UE is represented by $UE_x, x \in \{1, 2, \dots, M\}$ and K

Result: cluster head locations, $CH_i, i \in \{1, 2, \dots, K\}$

for $k \leftarrow 1$ to K **do**

 | $\mu_k \leftarrow$ some random location

end for

while *until converged* **do**

 | **for** $n \leftarrow 1$ to N **do**

 | $\pi_{nk} \leftarrow \operatorname{argmin}_k \|x_n - \mu_k\|^2$

 | **end for**

 | **for** $k \leftarrow 1$ to K **do**

 | $\mu_k \leftarrow \operatorname{mean}(\pi_{nk}, x_n)$

 | **end for**

end while

5. Numerical Results

The analysis is carried out in two phases. In the first, we keep MBS along with N UAV BSs and apply Q-learning algorithm, whereas in the second, we consider N UAV BSs to provide service to M UEs without considering UE association with MBS. Users are distributed in area spanning $1500 \text{ m} \times 1500 \text{ m}$, whereas the UAV BS can hold locations in the grid spanning $1500 \text{ m} \times 1500 \text{ m}$ with 150 m separation between neighbouring grid vertices. The output after running the Q-learning algorithm is compared to the best position obtained from brute-force search. The brute-force search reveals the reward for each possible state, corresponding to UAV BSs location and UEs association.

First, we consider three UAV BSs and an MBS providing service to 72 UEs. The agent is trained for 2000 episodes, 90 steps per episode, $\beta = 0.618$. The UEs are not associated to any UAV BS or MBS in the initial stage. Figure 2a shows the initial UAV BSs position for the system with three UAV BSs and an MBS. In the first step corresponding to episode 0, UAV BSs are placed at origin. Since the UAV BSs are not active, all the UEs are associated to the MBS. After running the Q-learning algorithm for 2000 episodes, the optimized UAV BSs position and UEs association is calculated.

Figure 2b shows the optimized UAV BSs position using the Q-Learning algorithm. However, the Q-learning algorithm does not reach the best UAV BSs position, which may be the case after numerous episodes. The UEs association is represented by providing the same colour code as that of the associated UAV BS or MBS. A circle representing coverage area of each UAV BS and MBS is also shown. It is important to note that a re iteration of RL from the initial stage may not result in same UAV BS positions as shown in the figure. However, a data rate or reward that is close to the value obtained with the present setting expected.

The UAV BSs position offering maximum data rate and corresponding UEs association found using brute force algorithm is shown in Figure 2c. This is equivalent to running the RL algorithm for large number of iterations or till infinity. The UAV BSs are positioned in such a way that the association of UEs with UAV BSs and MBS results in maximum data rate. The UEs are coloured according to their associated UAV BSs colour. The UEs associated with the MBS are also coloured in the same manner. It is worth noting that the index of UAV BSs may be different if iterations of the RL algorithm are carried out from the beginning. This is because the UAV BSs decides the next state based on exploratory actions, resulting in a random state. However, the maximum data rate of the system at the final stage will be same, even though there might be a change in index of UAV BSs.

Now, we consider a situation where only UAV BSs are used to provide service to UEs, which is illustrated in figures below. Figure 3a shows the UAV BSs' initial position for the system with three UAV BSs only. UAV BSs are placed at origin in the beginning of episode

0. Since there is UAV BSs–UEs association, the UEs are coloured black. The MBS in this system is inactive and does not serve any UEs.

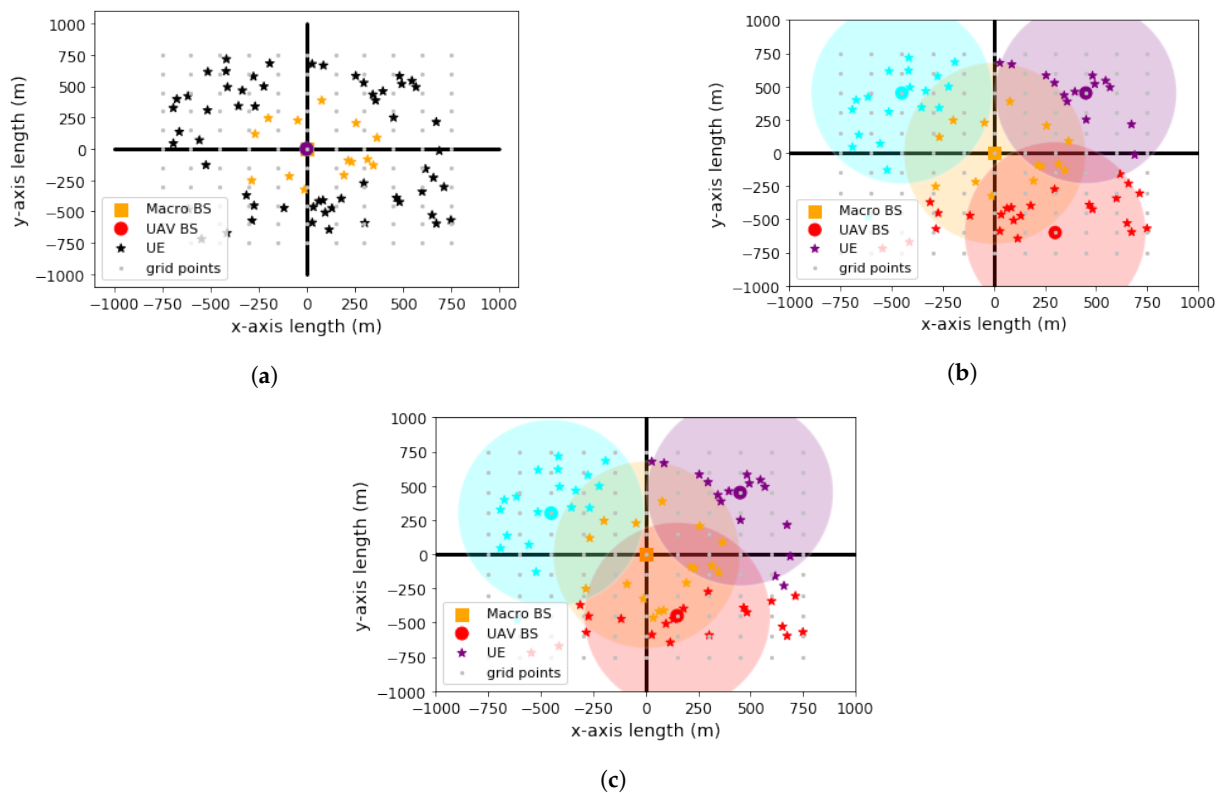


Figure 2. UAV positions and UEs assignment in a 3 UAV BS–MBS system. (a) Initial UAV BS positions; (b) Optimized UAV BSs positions; (c) Best UAV BSs positions.

After 2000 episodes of Q-learning, the optimized UAV BSs position and UEs association are given in Figure 3b. As discussed earlier, the Q-learning algorithm does not provide the best UAV BSs position after 2000 episodes. However, the performance is improved with each episode. The MBS do not serve any of the UEs compared to the previous case, and therefore, the data rate is less compared to the one with MBS. It is also worth noting that a reiteration of RL from the initial stage may not result in the UAV BS positions as shown in the figure. However, a data rate or reward that is close to the present value is expected.

The UAV BSs positions that offer maximum data rate and corresponding UEs association are found using brute-force algorithm and are shown in Figure 3c. On comparing with the system with MBS, this is also equivalent to running the RL algorithm for large number of iterations or till infinity. The UEs associated with the UAV BSs are also coloured with respect to the associated UAV BS. None of the UE is coloured with the colour corresponding to MBS, as the MBS is inactive in the system. The index of UAV BSs may be different if iterations of the RL algorithm are carried out from the beginning. It is due to the exploratory actions performed by UAV BSs, resulting in a random state. However, it is worth noting that the maximum data rate of the system at the final stage will be same, even though there might be a change in index of UAV BSs.

In Figure 4, positions of UAV BSs that provide the best data rate are calculated. The UAV BSs positions are found with UEs positions that were used to carry out RL based UAV BSs positioning. It is interesting to note that the K-means-based algorithm is applied to a system with no MBS, as MBS position cannot be changed. However, in a system with UAV BSs and UEs, the UAV BSs are placed at positions, where the UAV BSs were, after a large number of iterations in an RL based system. More importantly, the K-means algorithm proposes the UAV BSs positions that provides maximum data rate, without considering the path followed by the UAV BSs from an initial location.

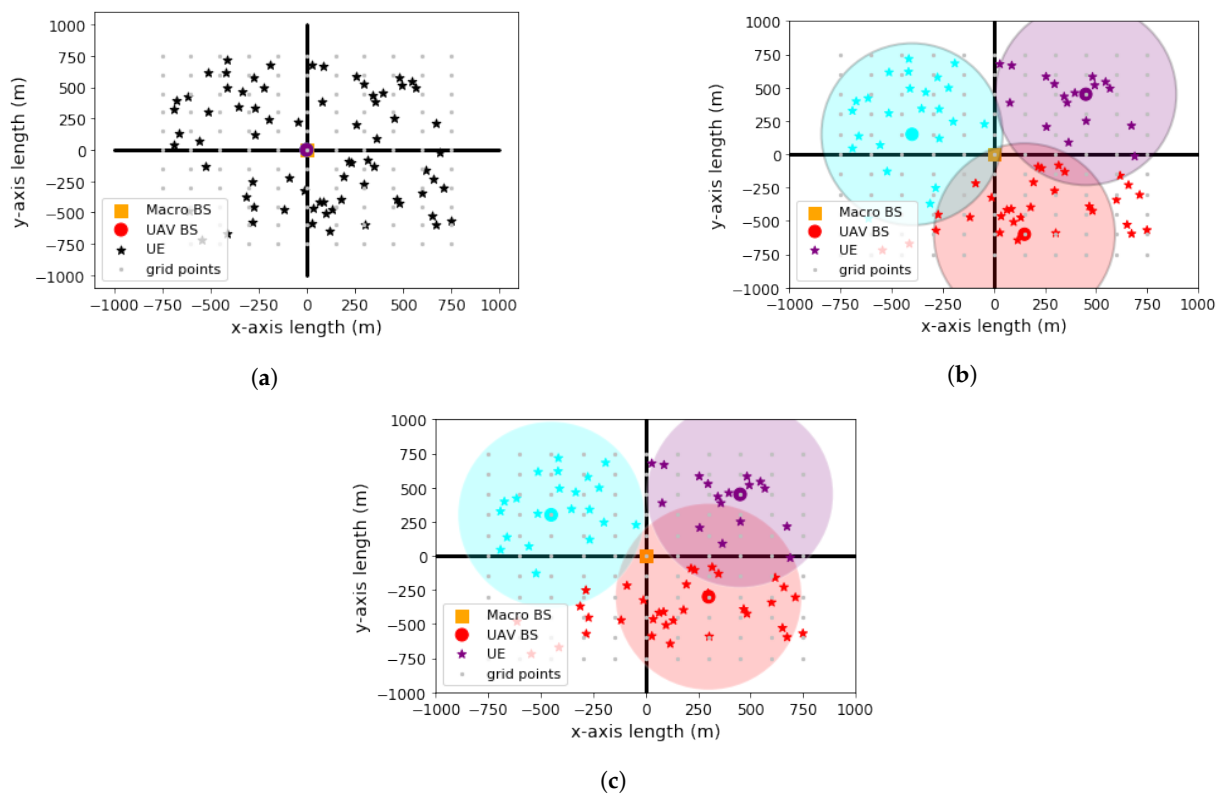


Figure 3. UAV positions and UEs assignment in a 3 UAV BS system. (a) Initial UAV BS positions; (b) Optimized UAV BS positions; (c) Best UAV BS positions.

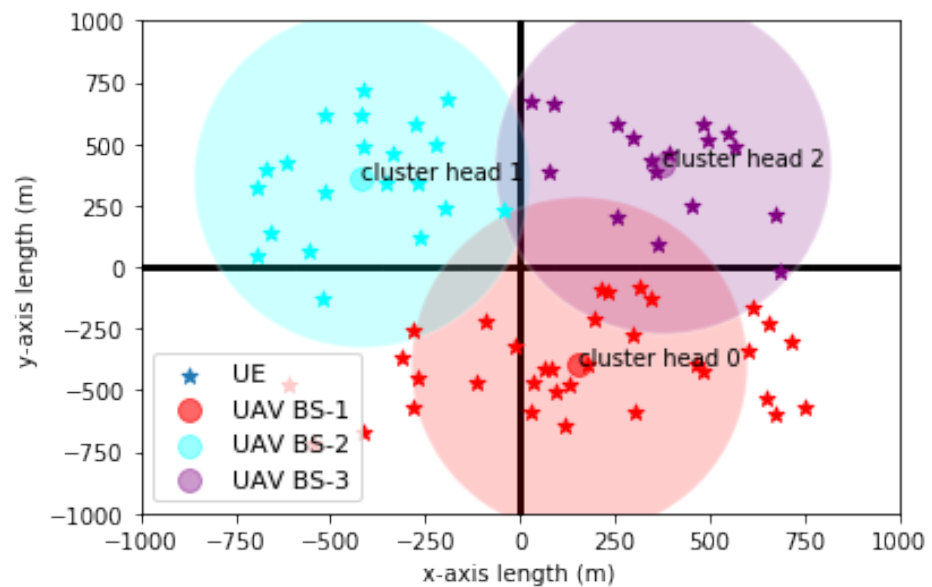


Figure 4. UAV BSs positioning using K-means algorithm.

Figure 5 explores the distribution of rewards in a 72 UE–3 UAV BSs–1 MBS system. The reward values are collected from a brute force search in the system. The plot follows a near-Gaussian distribution and has a maximum value corresponding to maximum reward as 21.1. The reward values correspond to data rates and use Equation (14) for conversion. From Figure 5, we can infer that the data rate obtained by RL algorithm after 2000 episodes is 69.2 % of the maximum possible data rate.

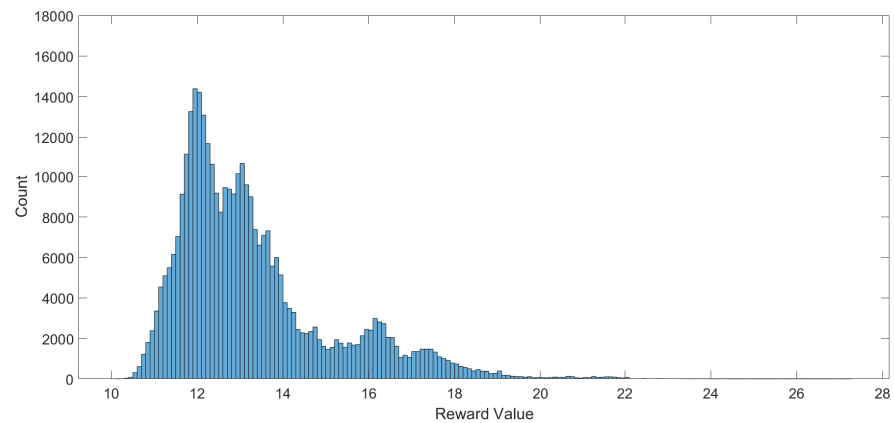


Figure 5. Histogram of rewards of 72 UE-3 UAV BS-1 MBS system

In Figure 6, the system with 72 UEs-3 UAV BSs system is considered, and a histogram of rewards is plotted using the values obtained from a brute force search for the same. Similar to Figure 5, the histogram follows a Gaussian distribution, and the maximum reward value is found to be 8.1. The reward values correspond to data rates and use Equation (14) for conversion. From Figure 6, it can be noted that the data rate obtained by RL algorithm after 2000 episodes is 60.2% of the maximum possible data rate.

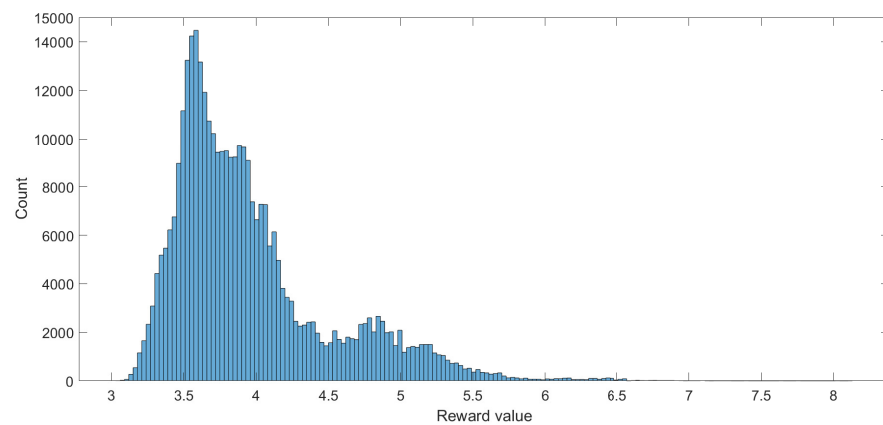


Figure 6. Histogram of reward of a 72 UEs-3 UAV BSs system.

Now, we evaluate the rewards obtained for the case with and without MBS. Figure 7 shows the total reward earned by the UEs-UAV BSs-MBS system. The total reward is the sum of all rewards of all s in an episode. That is, the sum of all rewards in each UAV BS position is plotted for each episode. The UAV BSs gather information and are trained with each episode. We can see that the agent learns and improves the total reward from each episode. The reward corresponding to the first episode is non-zero, as the UEs are associated with the MBS in the initial stage. A brute force algorithm is used to record data rate for every possible state. The data obtained using a brute force search reveal that the Q-learning approach with 2000 episodes provides a maximum sum-rate of 14.54 Mbits/s/Hz corresponding to 69.2% of the best possible reward.

Figure 8 shows the total reward versus episodes for 72 UEs-3 UAV BSs system. The total reward is the sum of all rewards observed in each iterations or UAV BS movements. The total reward of the agent increases rapidly compared to the case where MBS is used along with UAV BS. However, it can be noted that the reward is less compared to Figure 7. This is due to absence of the exclusive sub-channels dedicated to the MBS. After training the agent for 2000 episodes, the data rate corresponding to the optimized UAV BSs location is identified as 4.88 Mbits/s/Hz, which is 60.2 % of the best possible data rate.

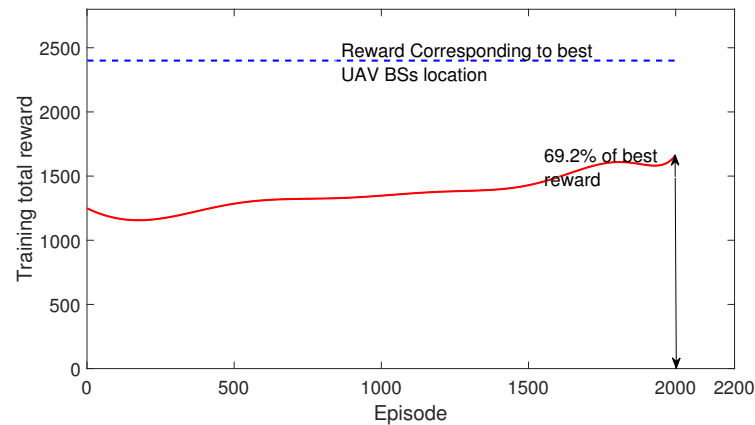


Figure 7. Reward for a 72 UEs-3 UAV BSs-1 MBS system.

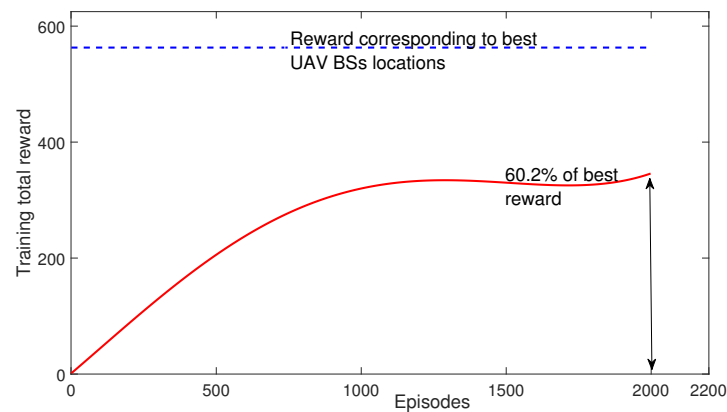


Figure 8. Reward for a 72 UEs-3 UAV BS system.

Rewards vs. ρ values are plotted for a 72 UEs-3 UAV BS system in Figure 9. Smaller ρ values correspond to lesser imperfection of user location information x , and the K-means algorithm provides better data rates due to better UAV BSs positioning. However, if ρ values are higher, due to imperfection of x values, UAV BSs are positioned in locations resulting in poor data rates. Hence, it is better to use an RL algorithm that is trained for large number of iterations in this case. In addition to dependency on UE positions, the K-means algorithm possesses another limitation, that is, the K-means algorithm does not consider the path the UAV BS follow to reach to the desired location.

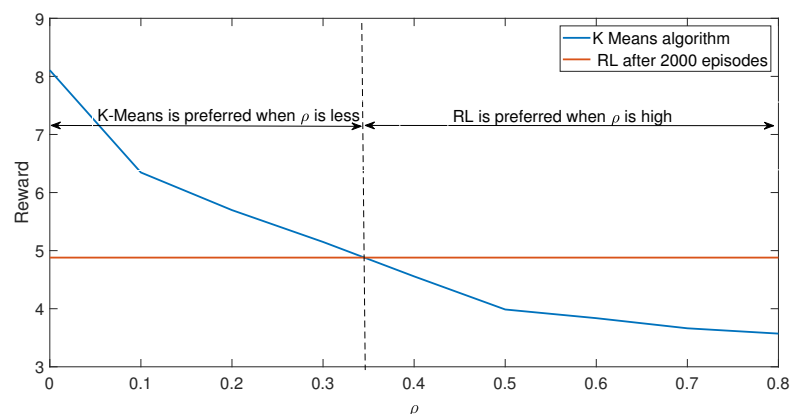


Figure 9. Rewards vs ρ plot of a 72 UEs-3 UAV BS system.

6. Discussion

An RL-aided UAV BS positioning architecture is proposed in this paper. The RL architecture improves the performance of the multiple UAV BS based communication system by exploratory movements, which can lead to collision and movement of UAV BS beyond the intended search area. The paper considers practical deployment of UAV BS to maximize downlink data rate, while protecting the UAV BS by avoiding UAV BS collision and by avoiding UAV BS to explore beyond the prescribed search area. To the best of our knowledge, such an attempt is not made in works involving UAV BSs. RL-based UAV BS deployment is considered in [24–28] to improve performance of system. However, in [24], authors use single UAV BS, where there is no possibility of collision between other UAV BSs and in [25,26], UAV BSs are only used as backhaul to serve UAV BSs. In [27], the authors consider the deep deterministic policy gradient algorithm, which is a neural network-based algorithm. Work in [28] is an extension of epsilon-greedy algorithm, which is a basic learning scenario in RL, making it an unreal scenario for a UAV BS-based network. Though several studies [24–28] use RL to optimize UAV BS, they do not address the issue of UAV BSs collision and movement of UAV BS beyond the intended search area. In our paper, we not only address these issues, but also produce an RL system to perform more exploratory movements initially and perform exploitative movements in final stages to improve sum-reward.

In this paper, we consider two scenarios, where the first uses MBS along with UAV BSs, and later, we use only UAV BSs. Figures 2b and 3b reveal that the UAV BSs are not damaged due to collision or loss due to exploratory movements outside the intended search area. It is worth noting that the system consist of multiple UAV BS and MBS serving multiple UE; therefore, a comparison with [24–28] in terms of data rate vs. episodes may not be meaningful, as the underlying system architecture and assumptions are different, as discussed before. Figures 7 and 8 show the improvement in data rate with RL-assisted UAV BS position optimization. On comparing it with brute force algorithm, where the data rate for each possible UAV BSs position is noted, the results reveal that the system with MBS reaches 69.2% of the best possible data rate. Similarly, the system with UAV BSs reaches 60.2% of the best possible data rate. Finally, the RL algorithm is compared with the K-means algorithm with imperfect UE locations information. Results reveal that the RL performs better when imperfection in UE locations are more.

The RL architecture considered in the paper performs the optimization in a centralized manner, which means that the UAV BS is not deciding the next movement by itself. This is a limitation, and a decentralized UAV BS position optimization is a possible future work.

7. Conclusions

In this paper, we propose an RL-aided UAV BSs positioning approach for overall downlink data rate maximization in HetNets with possible coexisting macro BSs. By exploiting the Q-learning algorithm, the UAV BSs position themselves in optimal locations without collision. Considering the user distribution in area spanning 1500 m × 1500 m, after 2000 episodes, the cellular system with 3 UAV BSs and an MBS reaches 69.2% of the best possible data rate. Similarly, the system with 3 UAV BSs reaches 60.2% of the best possible data rate. The proposed approach makes use of a learning process that explores more in initial stages, without expecting maximum reward, and then, in later stages, it takes actions to maximize it. The RL algorithm is compared with the K-means algorithm with imperfect UE locations information, and it turns out that the RL performs better when imperfection in UE locations are more.

Author Contributions: Conceptualization, S.P.G. and M.M.; methodology, S.P.G.; software, S.P.G.; validation, S.P.G.; writing—original draft preparation, S.P.G.; writing—review and editing, S.P.G. and M.M.; visualization, S.P.G.; supervision, M.M.; project administration, M.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Erdelj, M.; Natalizio, E.; Chowdhury, K.R.; Akyildiz, I.F. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervasive Comput.* **2017**, *16*, 24–32. [[CrossRef](#)]
2. Song, Q.; Zeng, Y.; Xu, J.; Jin, S. A survey of prototype and experiment for UAV communications. *Sci. China Inf. Sci.* **2021**, *64*, 1–21. [[CrossRef](#)]
3. Tariq, F.; Khandaker, M.R.; Wong, K.K.; Imran, M.A.; Bennis, M.; Debbah, M. A speculative study on 6G. *IEEE Wirel. Commun.* **2020**, *27*, 118–125. [[CrossRef](#)]
4. Ueyama, J.; Freitas, H.; Faical, B.S.; Geraldo Filho, P.; Fini, P.; Pessin, G.; Gomes, P.H.; Villas, L.A. Exploiting the use of unmanned aerial vehicles to provide resilience in wireless sensor networks. *IEEE Commun. Mag.* **2014**, *52*, 81–87. [[CrossRef](#)]
5. Zhang, L.; Celik, A.; Dang, S.; Shihada, B. Energy-Efficient Trajectory Optimization for UAV-Assisted IoT Networks. *IEEE Trans. Mob. Comput.* **2021**. [[CrossRef](#)]
6. Chen, M.; Saad, W.; Yin, C. Liquid state machine learning for resource allocation in a network of cache-enabled LTE-U UAVs. In Proceedings of the GLOBECOM 2017—2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6.
7. Challita, U.; Ferdowsi, A.; Chen, M.; Saad, W. Machine learning for wireless connectivity and security of cellular-connected UAVs. *IEEE Wirel. Commun.* **2019**, *26*, 28–35. [[CrossRef](#)]
8. Li, B.; Fei, Z.; Zhang, Y. UAV communications for 5G and beyond: Recent advances and future trends. *IEEE Internet Things J.* **2018**, *6*, 2241–2263. [[CrossRef](#)]
9. Bithas, P.S.; Michailidis, E.T.; Nomikos, N.; Vouyioukas, D.; Kanatas, A.G. A survey on machine-learning techniques for UAV-based communications. *Sensors* **2019**, *19*, 5170. [[PubMed](#)]
10. Cicek, C.T.; Gultekin, H.; Tavli, B.; Yanikomeroglu, H. UAV base station location optimization for next generation wireless networks: Overview and future research directions. In Proceedings of the 2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS), Muscat, Oman, 5–7 February 2019; pp. 1–6.
11. D’Andrea, C.; Garcia-Rodriguez, A.; Geraci, G.; Giordano, L.G.; Buzzzi, S. Analysis of UAV Communications in Cell-Free Massive MIMO Systems. *IEEE Open J. Commun. Soc.* **2020**, *1*, 133–147. [[CrossRef](#)]
12. Zhang, S.; Shi, S.; Feng, T.; Gu, X. Trajectory planning in UAV emergency networks with potential underlying D2D communication based on K-means. *EURASIP J. Wirel. Commun. Netw.* **2021**, 1–19. [[CrossRef](#)]
13. Wu, X.; Wei, Z.; Cheng, Z.; Zhang, X. Joint optimization of UAV Trajectory and User Scheduling Based on NOMA Technology. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea, 25–28 May 2020; pp. 1–6.
14. Alsamhi, S.H.; Almalki, F.; Ma, O.; Ansari, M.S.; Lee, B. Predictive Estimation of Optimal Signal Strength from Drones over IoT Frameworks in Smart Cities. *IEEE Trans. Mob. Comput.* **2021**, *1*. [[CrossRef](#)]
15. Amorim, R.; Wigard, J.; Nguyen, H.; Kovacs, I.Z.; Mogensen, P. Machine-learning identification of airborne UAV-UEs based on LTE radio measurements. In Proceedings of the 2017 IEEE Globecom Workshops (GC Wkshps), Singapore, 4–8 December 2017; pp. 1–6.
16. El Hammouti, H.; Benjillali, M.; Shihada, B.; Alouini, M.S. A distributed mechanism for joint 3D placement and user association in UAV-assisted networks. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April 2019; pp. 1–6.
17. Bor-Yaliniz, R.I.; El-Keyi, A.; Yanikomeroglu, H. Efficient 3-D placement of an aerial base station in next generation cellular networks. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 23–27 May 2016; pp. 1–5.
18. Lyu, J.; Zeng, Y.; Zhang, R.; Lim, T.J. Placement optimization of UAV-mounted mobile base stations. *IEEE Commun. Lett.* **2016**, *21*, 604–607. [[CrossRef](#)]
19. Alzenad, M.; El-Keyi, A.; Yanikomeroglu, H. 3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements. *IEEE Wirel. Commun. Lett.* **2017**, *7*, 38–41. [[CrossRef](#)]
20. Zeng, Y.; Zhang, R. Energy-efficient UAV communication with trajectory optimization. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 3747–3760. [[CrossRef](#)]
21. Du, J.; Jiang, C.; Wang, J.; Ren, Y.; Debbah, M. Machine Learning for 6G Wireless Networks: Carrying Forward Enhanced Bandwidth, Massive Access, and Ultra reliable/Low-Latency Service. *IEEE Veh. Technol. Mag.* **2020**, *15*, 122–134. [[CrossRef](#)]
22. Lai, C.C.; Wang, L.C.; Han, Z. Data-driven 3D placement of UAV base stations for arbitrarily distributed crowds. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6.
23. Cheng, F.; Zou, D.; Liu, J.; Wang, J.; Zhao, N. Learning-Based User Association for Dual-UAV Enabled Wireless Networks With D2D Connections. *IEEE Access* **2019**, *7*, 30672–30682. [[CrossRef](#)]
24. Ghanavi, R.; Kalantari, E.; Sabbaghian, M.; Yanikomeroglu, H.; Yongacoglu, A. Efficient 3D aerial base station placement considering users mobility by reinforcement learning. In Proceedings of the IEEE Wireless Communications and Networking Conference, Barcelona, Spain, 15–18 April 2018; pp. 1–6.

25. Fotouhi, A.; Ding, M.; Giordano, L.G.; Hassan, M.; Li, J.; Lin, Z. Joint Optimization of Access and Backhaul Links for UAVs Based on Reinforcement Learning. In Proceedings of the 2019 IEEE Globecom Workshops (GC Wkshps), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6.
26. Fotouhi, A.; Ding, M.; Hassan, M. Dynamic base station repositioning to improve performance of drone small cells. In Proceedings of the 2016 IEEE Globecom Workshops (GC Wkshps), Washington, DC, USA, 4–8 December 2016; pp. 1–6.
27. Yin, S.; Zhao, S.; Zhao, Y.; Yu, F.R. Intelligent trajectory design in UAV-aided communications with reinforcement learning. *IEEE Trans. Veh. Technol.* **2019**, *68*, 8227–8231. [[CrossRef](#)]
28. Bayerlein, H.; De Kerret, P.; Gesbert, D. Trajectory optimization for autonomous flying base station via reinforcement learning. In Proceedings of the 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 25–28 June 2018; pp. 1–5.
29. Bokani, A.; Hassan, M.; Kanhere, S.; Zhu, X. Optimizing HTTP-based adaptive streaming in vehicular environment using markov decision process. *IEEE Trans. Multimed.* **2015**, *17*, 2297–2309. [[CrossRef](#)]
30. Liu, J.; Tao, X.; Lu, J. Mobility-Aware Centralized Reinforcement Learning for Dynamic Resource Allocation in HetNets. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6.
31. Yan, C.; Fu, L.; Zhang, J.; Wang, J. A comprehensive survey on UAV communication channel modeling. *IEEE Access* **2019**, *7*, 107769–107792. [[CrossRef](#)]