Q-Learning-based Setting of Cell Individual Offset for Handover of Flying Base Stations

Aida Madelkhanova†, Zdenek Becvar†, and Thrasyvoulos Spyropoulos*
†Dpt. of Telecommunication Engineering, FEE, Czech Technical University in Prague, Prague, Czech Republic
* Communication Systems Department, EURECOM, Sophia Antipolis, France
emails: † {aida.madelkhanova, zdenek.becvar}@fel.cvut.cz, * thrasyvoulos.spyropoulos@eurecom.fr

***Abstract*—Flying base stations (FlyBSs) are widely used to improve coverage and/or quality of service for users in mobile networks. To ensure a seamless mobility of the FlyBSs among the static base stations (SBSs), an efficient handover mechanism is required. We focus on the handover of FlyBSs among SBSs and we dynamically adjust the cell individual offset (CIO) of the SBSs based on their load to increase the sum capacity of the users served by the FlyBSs while considering also a handover cost. Due to complexity of the defined problem and limited knowledge of other parameters required for conventional optimization methods, we adopt Q-learning to solve the problem. For Q-learning, we define a reward function reflecting the trade-off between the capacity of users and the cost of performed handovers. The proposed Q-learning based approach converges promptly and increases the sum capacity of the users served by the FlyBSs by up to 23% for eight deployed FlyBSs comparing to state-of-the-art algorithms. At the same time, the number of handovers performed by the FlyBSs is notably reduced (up to 25%) by the proposal.**

***Index Terms*—Flying base station, handover, cell individual offset, reinforcement learning.**

I. Introduction

The flying base stations (FlyBSs), represented by the unmanned aerial vehicles (UAVs) carrying a hardware for wireless communication, are seen as a suitable solution for the future mobile networks due to a flexible deployment and a high mobility. The FlyBSs allow to extend the network coverage or to boost quality of service in a specific area [1]. Due to a fast and prompt deployment, the FlyBSs are suitable for emergency situations or short-time events [2]. However, an integration of the FlyBSs to the mobile networks introduces new challenges, such as finding an optimal position of the FlyBS [3], optimizing the FlyBS's trajectory and power allocation [4], or an association of user equipments (UEs) to the FlyBSs [5].

Another key challenge is related to a seamless mobility of the FlyBSs among the static base stations (SBSs) [6]. The FlyBSs move over time and their trajectory is arbitrary and hard to be predicted, as the FlyBSs are positioned according to the movement of the served UEs. The arbitrary trajectory can lead to rapid changes in the quality of channels between the FlyBS and the served UEs as well as between the FlyBS and the static base station (SBS) that provides connectivity of the FlyBS to the network. These rapid changes in the channel quality can result in unpredictable and frequent handovers of the FlyBSs among the SBSs and, consequently, to a handover failure, packet losses, and/or overloading of some SBSs. Therefore, an efficient handover mechanism of the FlyBSs among the SBSs is required to provide a reliable communication of the UEs through the FlyBSs.

In conventional mobile networks, the handover of the UE between the SBSs is commonly initiated when a target SBS (i.e., the base station to which the handover should be performed) provides a channel of a higher quality than the current serving SBS. To avoid frequent handovers and/or handover failures, the decision on the handover is tuned via control parameters, such as, a hysteresis, a time-to-trigger (TTT), or cell individual offset (CIO) (please refer to [7] for more details). Thus, the handover usually takes place when the target SBS provides the channel of a quality that is at least the hysteresis and/or the offset(s) above the quality of the channel to the serving SBS for an interval of the TTT.

In the conventional mobile networks without the FlyBSs, the optimization of the hysteresis, TTT, and the offsets is heavily addressed. For example, in [8], the authors propose an adaptation of the hysteresis according to relative qualities of the channels from the serving and neighboring SBSs. This approach reduces the number of handovers; however, it does not improve the UEs' throughput. In [9], the authors adapt the hysteresis via fuzzy logic to minimize the number of performed handovers. However, an impact of the handover on the throughput of the UEs is not considered. A reactive load balancing algorithm based on reinforcement learning is developed in [10]. The reactive algorithm is based on an adaptation of the CIO. The CIO is a cell-specific handover control parameter enabling to control the association of the UEs with the SBSs and to regulate a load of the SBSs. In general, the CIO of the overloaded SBS is set to a lower value and the CIO of the target SBS is increased. The authors adjust the CIO of the serving and neighboring SBSs by a specific value so that the offset for the serving and neighboring SBSs is of the same absolute value, but opposite sign (e.g. -0.5 dB and +0.5 dB for the serving and neighboring SBSs, respectively). The CIO adjustment in [10] is based on a distribution of the cell-edge UEs in the area. However, the UEs' distribution is typically unknown in practical scenarios. In [11], the authors adjust CIO for the load balancing purposes. Three predetermined

This work has been supported by Grant No. LTT 20004 funded by Ministry of Education, Youth and Sports and by the grant of Czech Technical University in Prague No. SGS20/169/OHK3/3T/13.

thresholds are defined to distinguish four levels of the SBSs' load. A higher CIO is selected for the SBS with a lower load, and a lower CIO is set for the highly loaded SBSs. This CIO adjustment relieves the heavy traffic load of the SBS; however, it does not improve the UEs' throughput. In [12], a UE association algorithm based on reinforcement-learning is proposed to reduce the number of handovers in the network with FlyBSs and also a FlyBS mobility control algorithm is adopted to optimize the system throughput. The proposed algorithm reduces the number of handovers performed by the UEs, while increasing the throughput of the system; however, the backhaul links with the SBSs are not taken into account.

Few works study the problem of handover in the networks with the UEs represented by the UAVs. The handover management for the UAV acting as the UE via a dynamic adjustment of the SBSs' antenna tilt angles is outlined in [13]. The authors demonstrate that an intelligent antenna tilting reduces a handover rate for a simple mobility scenario with the UAV UE traveling along a linear trajectory. In [14], the authors propose a scheme adjusting the handover parameters for the UAVs UEs and making the handover decision based on the UAV's trajectory to reduce the number of performed handovers. In [15], the handover based on the reinforcement learning is proposed to maximize the received signal quality at the UAV UEs while minimizing the number of handovers performed by these UAV UEs. Despite the encouraging results, the works [13] - [15] assume the scenario with a predefined and a priori known trajectory of the UAV UEs. This assumption is, however, not valid in the scenario with the UAVs acting as the FlyBSs serving the moving UEs, since the trajectory of the FlyBSs is unknown and depends on the UEs' movement. In our prior work [16], we investigate the CIO adjustment in the scenario with multiple SBSs, but just single FlyBS. The proposed reinforcement learning based algorithm adjusts CIO according to the load of SBSs to maximize the sum capacity of the UEs served by the FlyBS. However, a cost of handovers (e.g., signaling overhead and related additional energy consumption) is not taken into account, hence, the solution can lead to redundant handovers.

None of the existing papers targets an optimization of the handover of multiple FlyBSs to maximize the sum capacity while also avoiding an excessive number of performed handovers. Hence, in this paper, we propose a framework for a management of handovers of the FlyBSs among the SBSs to maximize the sum capacity of the users in the scenario with multiple FlyBSs taking the handover cost into account to avoid redundant handovers. Due to a dynamic nature and an unpredictable behavior of the UEs (and consequently also FlyBSs serving these UEs) we employ Q-learning to set a proper CIO for individual SBSs. The CIO is dynamically adjusted using Q-learning to provide an efficient mobility support in the sky. To adapt Q-learning to our targeted problem, we introduce a new reward function taking the impact of handover on the sum capacity as well as the handover cost into account. Moreover, we consider the communication load implied by the FlyBSs (i.e., load generated by the UEs served by individual

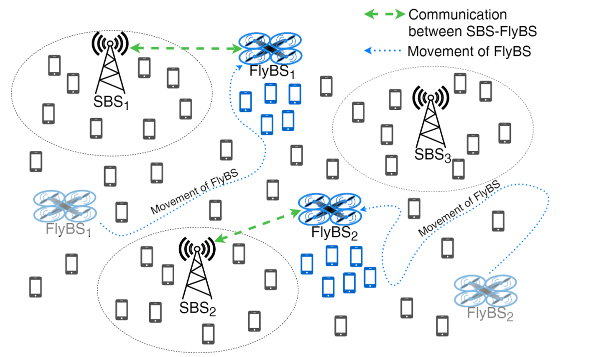


Fig. 1: System model for handover of multiple FlyBSs serving mobile UEs among SBSs.

FlyBSs) in the process of CIO determination. This allows to avoid overloading of the SBSs and, consequently, to prevent unnecessary handovers caused by SBSs' overloading. The proposed solution leads to a notable increase in the UEs' sum capacity and a decrease in the number of handovers compared to state-of-the-art solutions including our prior work [16].

The rest of this paper is organized as follows. Section II presents the system model and defines the problem addressed in this paper. Then, in Section III, we present our proposed reinforcement learning-based determination of the CIO for the SBSs. The simulation results and their discussion are provided in Section IV. Section V concludes the paper.

II. System model and problem formulation

In this section, we first outline the system model considered in this paper. Then, we formulate the targeted problem.

A. System model

We assume N UEs deployed in the area covered with K_S conventional SBSs and additional K_F FlyBSs. In total, there are $K = K_S + K_F$ base stations (BSs). Note that the label BS represents both the SBSs and the FlyBSs in this paper. All UEs in the system require a certain communication capacity c_{req}. For a clarity of the following explanations and presentation of results, we assume the same c_{req} for all UEs. However, our proposed solution is suitable for any, even diverse, c_{req} of individual UEs. Out of N UEs, N_u UEs cannot receive c_{req} from the SBSs (e.g., due to a high load of the SBSs). These N_u UEs, denoted as uncovered UEs, are connected to the network via the FlyBSs, which relay the communication from the adjacent SBS to the uncovered UEs.

The position of the n-th UE changes over time and the FlyBSs follow the uncovered UEs. To maintain a reliable connectivity, each FlyBS performs handovers during flight and, thus, the association of the FlyBSs to the SBSs changes as well over time. We define a binary parameter $\beta_{k,f}^{FlyBS}$ to indicate if the f-th FlyBS is associated to the k-th SBS ($\beta_{k,f}^{FlyBS}$= 1), or not ($\beta_{k,f}^{FlyBS}$= 0). Like in [17], the position of each FlyBS corresponds to the center of gravity of all UEs associated to this FlyBS. Note that our proposed solution for the CIO adjustment does not depend on the positions of the

FlyBSs and can be applied together with any other approaches for the determination of the FlyBSs' positions.

In our model, we consider the downlink communication from the SBSs to the UEs either directly or via the FlyBSs. The signal to interference plus noise ratio (SINR) $\gamma_{f,n}$ observed by the n-th UE from the f-th FlyBS is defined as:

$$\gamma_{f,n} = \frac{P_f h_{f,n}}{\sum_{i=1, i\neq f}^{K} P_i h_{i,n} + \sigma^2} \tag{1}$$

where P_f is the transmission power of the f-th FlyBS serving the n-th UE, $h_{f,n}$ is the channel gain between the n-th UE and the f-th FlyBS, the term $\sum_{i=1,i\neq f}^{K} P_i h_{i,n}$ represents the co-channel interference from other BSs, P_i is the transmission power of the i-th BS representing the interference to the n-th UE, $h_{i,n}$ corresponds to the channel gain between the n-th UE and the i-th interfering BS, and σ^2 represents the noise.

Similarly, the SINR at the f-th FlyBS receiving data from the k-th serving SBS is expressed as:

$$\gamma_{k,f} = \frac{P_k h_{k,f}}{\sum_{i=1, i\neq k}^{K} P_i h_{i,f} + \sigma^2} \tag{2}$$

where $h_{k,f}$ is the channel gain between the k-th serving SBS and the f-th FlyBS, $\sum_{i=1,i\neq k}^{K} P_i h_{i,f}$ represents the interference from other BSs, and $h_{i,f}$ stands for the channel gain between the i-th interfering BS and the f-th FlyBS.

As we adopt decode and forward relaying, the relaying channel capacity for the communication of the the n-th UE via the f-th FlyBS is defined as [18]:

$$c_n = \frac{B_n}{2} min\{log_2(1+\gamma_{k,f}), log_2(1+\gamma_{f,n})\} \tag{3}$$

where B_n denotes the bandwidth of the n-th UE's channel.

The k-th BS serves a set of the UEs generating the load ρ_k to this BS. The load is defined as the ratio of the bandwidth allocated to the UEs associated to the k-th BS versus the total amount of bandwidth available for the given BS, i.e.:

$$\rho_k = \frac{\sum_{n=1}^{N} \beta_{k,n}^{UE} B_n}{B} \tag{4}$$

where the binary parameter $\beta_{k,n}^{UE} \in \{0,1\}$ indicates if the n-th UE is associated to the k-th BS ($\beta_{k,n}^{UE} = 1$), or not ($\beta_{k,n}^{UE} = 0$), and B is the total bandwidth available for the k-th BS.

As the FlyBSs follow the moving UEs, each FlyBS can perform handover(s) among the SBS during the flight. Like common UEs in the mobile networks, also the FlyBS measures the channel quality from the neighboring SBSs. The channel quality measurement report is periodically sent to the serving SBS in a similar way as the common UEs report their channel quality in the mobile networks [13]. Based on the measurement results, the serving SBS decides to handover the FlyBS to one of the neighboring SBSs if a higher signal quality can be reached. We consider the handover based on the A3 event [7] that involves the hysteresis, TTT, and CIO. The channel quality is represented by the received signal strength (RSS) expressed as $RSS_{k,f} = P_k h_{k,f}$. The FlyBS performs the handover to the target SBS if the target SBS satisfies the following condition for more than the period of TTT:

$$RSS_t + CIO_t - Hys > RSS_s + CIO_s \tag{5}$$

where the indices s and t correspond to the parameters of the serving and target SBSs, respectively, and Hys is the value of hysteresis parameter in dB.

B. Problem formulation

In this paper, we focus on the handover of the multiple moving FlyBSs among the SBSs. The objective is to adjust CIOs of the SBSs so that the sum capacity of the UEs served by the FlyBSs is maximized. However, using the sum capacity as a sole objective could lead to an excessive number of redundant handovers resulting in an additional signaling overhead increasing an energy consumption, which is a critical factor for the FlyBSs. Thus, the number of handovers is also accounted for to enable affordable cost for network operation. Consequently, the targeted problem is formally defined as:

$$\boldsymbol{CIO} = \underset{\boldsymbol{CIO}\in O}{\text{argmax}} \sum_{n=1}^{N_u} c_n - \mu \tag{6}$$

$$\text{subject to} \sum_{k=1}^{K} \beta_{k,n}^{UE} = 1, \ \forall n \in \langle 1, N \rangle, \tag{6a}$$

$$\sum_{k=1}^{K_S} \beta_{k,f}^{FlyBS} = 1, \ \forall f \in \langle 1, K_F \rangle, \tag{6b}$$

$$\sum_{n=1}^{N} \beta_{k,n}^{UE} B_n \leqslant B, \ \forall k \in \langle 1, K_S \rangle. \tag{6c}$$

where $O = \langle CIO_{min}, CIO_{max} \rangle$ defines the set of possible CIO values ranging from CIO_{min} to CIO_{max} and μ denotes the handover cost. The constraint (6a) ensures that each UE is associated to just one BS and the constraint (6b) ensures that each FlyBS is associated to just one SBS. Furthermore, the constraint (6c) guarantees that the SBSs do not allocate more bandwidth than available.

III. Proposed CIO adjustment based on Q-learning

The problem defined in (6) is complex given a high randomness of the network environment caused by the mobility of both the UEs and the FlyBSs. This problem can be solved by non-linear optimization techniques. However, these optimization-based techniques require the exact knowledge of the network state including locations of the UEs and the FlyBSs, which might not always be available. Moreover, even with the perfect information of all relevant parameters, such optimization problem is NP-hard due to its definition as a non-convex function and difficult to solve efficiently. Thus, we adopt the reinforcement learning to adjust CIO of the SBSs for handover decision of the FlyBSs. Unlike other machine learning algorithms, which exploit historical data, the reinforcement learning allows the network to learn and improve its decision by interacting with an unknown environment through time. We propose the Q-learning-based algorithm to obtain the optimal

CIO adjustment policy for the serving as well as target SBSs. In this section, we first briefly introduce the background in reinforcement learning related to our targeted problem, then, we present the proposed CIO adjustment scheme.

A. Preliminaries on Reinforcement Learning

In the reinforcement learning, an agent interacts with an environment based on a set of given actions. Reinforcement learning is often described via Markov decision process (MDP) characterized by a tuple consisting of (S, A, P, R), where S and A denote the sets of all possible states and actions, respectively, P denotes the transition probabilities for the states when a particular action is taken, and R is the reward function [19]. At each state, the MDP takes the action that maximizes the expected sum of discounted future rewards. In many practical reinforcement learning problems, the models defining the P and R are not available. In these problems, the optimal policy can still be derived using a model-free reinforcement learning algorithm known as Q-learning. The Q-learning is typically simpler and more flexible to implement than the model-based algorithms, since the dynamics or the model of the environment are not required to be known a priori. Assuming π is the policy of choosing the actions, the Q-value $Q^{\pi}(s,a)$ for every state-action pair indicates how good the action a performed in that state s is. Using an iterative process, the agent eventually learns the optimal Q-values $Q(s,a)$ over time. When the agent performs the action a_t in the state s_t at the time t, an immediate reward r_t is received and the agent transits to the state s_{t+1}. The new Q-value is evaluated using:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left[r_t + \lambda max Q(s_{t+1},a) - Q(s_t,a_t)\right], \tag{7}$$

where $\alpha \in (0,1]$ is the learning rate that balances new information against previous knowledge, and $\lambda \in (0,1)$ is the discount factor that balances between the immediate and future rewards. We adopt the ϵ-greedy policy, where the agent tries to obtain the highest reward at each training step, but also checks for other actions, which can improve the estimated future reward [19]. The learning starts with ϵ = 1, then, ϵ is continuously reduced to ϵ = 0 via multiplication by the decay factor η = 0.99 at each learning step [13].

B. Q-learning for CIO adjustment

The objective defined in (6) is interpreted as the problem, where the agent maximizes its final cumulative reward by interacting with an unknown environment over time. The FlyBS is usually constrained with a limited energy and any additional energy consumption is not welcome. Thus, we assume that the network is equipped with a central agent, e.g., in an edge server or a (software) entity in the operator's core network, that can monitor the load of the BSs and implement Q-learning. However, the proposed solution does not depend on the agent representation and can be applied also in the case when each FlyBS acts as an agent.

The sets of states and actions and the reward function for our targeted problem are defined as follows. The

Algorithm 1 Q-learning for CIO adjustment

1: **Input:** number of SBSs and FlyBSs, BSs' load, action set (possible CIO values)
2: initialize $Q(s,a)$ and $S(1)$
3: **for** each learning step t **do**
4:     **for** each FlyBS f **do**
5:         choose action $A(t)$ using ϵ-greedy policy
6:         perform action $A(t)$ and update CIOs for SBSs
7:         **if** A3 event handover trigger condition (5) satisfied
8:             perform handover
9:         **end if**
10:        calculate $r(t)$ using (8)
11:        update Q-table using (7)
12:        $S(t) \leftarrow S(t+1)$
13:    **end for**
14: **end for**
15: **Output:** Q-table

state of the network is represented by the load of the SBSs. Thus, the set of states $S(t)$ is defined as $S(t) = [\rho_1(t), \rho_2(t), \ldots, \rho_k(t), , \rho_f(t)]$, where $\rho_k(t) \in [0,1]$ corresponds to the load of the k-th SBS at the time t, and $\rho_f(t) \in [0,1]$ corresponds to the load generated by the f-th FlyBS (i.e., by the UEs served by this FlyBS) at the time t. The action is understood as a selection of the CIO for each SBS. Thus, the agent determines the CIOs for the SBSs via a selection of suitable actions $A(t) = [CIO_1(t), CIO_2(t), \ldots, CIO_k(t)]$, where the $CIO_k(t)$ corresponds to the CIO of the k-th SBS at the time t. The values of $CIO_k(t)$ are selected from the discrete set of $\langle CIO_{min}, CIO_{max}\rangle$ dB of a size L. As we target the maximization of the sum capacity of the N_u UEs served by the FlyBSs (see (6)) and avoiding the redundant handovers by taking into account the cost associated to handover events, the reward function $r(t) \in \langle 0,1\rangle$ at the time t is defined as:

$$r(t) = \frac{1}{c_{req}}\left(\frac{\sum_{n=1}^{N_u} c_n(t)}{N_u} - n_h\mu_u\right), \tag{8}$$

where $c_n(t)$ is the channel capacity of the n-th UE served by the FlyBS at the time t, the term $n_h\mu_u$ represents the handover cost μ, n_h is the number of UEs served by the FlyBS performing handover, and μ_u denotes the handover cost for the UE.

The pseudo-code of the proposed Q-learning process is presented in Algorithm 1. First, the Q-table is initiated with random values from interval $(0,1)$ (see line 2 in Algorithm 1). Based on the current state $S(t)$, the ϵ-greedy policy is performed to choose either the random or optimal action (see line 5). Afterwards, the chosen action is performed (line 6). If A3 handover trigger condition is satisfied (lines 7-9), the FlyBS performs the handover from the serving SBS to the target SBS. Then, the reward of the handover is calculated according to (8) (line 10). Finally, values for selecting different actions are updated in the Q-table according to (7) (line 11).

TABLE I: Simulation Parameters

Parameter	Value
Simulation area	1000m × 1000m
Carrier frequency	2 GHz
Tx power of SBS/FlyBS	23/15 dBm
Bandwidth of SBS	100 MHz
SBS/FlyBS/UE height	30/80/1.5 m
Number of UEs	150
Hysteresis margin	3 dB
Time step	1 s
CIO set	{-6, -3, 0, 3, 6} dB [22]

IV. Performance Evaluation

In this section, we evaluate the performance of our proposed Q-learning-based CIO adjustment scheme via simulations in MATLAB. We consider a suburban scenario with the simulation area of 1000×1000 m. Within this area, three conventional SBSs are deployed randomly with a minimum inter-site distance of 500 m (see Figure 1). Furthermore, up to eight FlyBSs are placed in the simulation area. The position of each FlyBS corresponds to the center of gravity of all UEs associated to this FlyBS [17]. Note that the proposed solution can be applied together with any other approach of the FlyBSs' positioning. The BSs serve 150 UEs moving with a random speed varying between 1.0 and 1.8 m/s. Out of all UEs, 60 UEs are randomly distributed and deployed uniformly around the SBSs within a circle with a radius of 150 m. Another 30 UEs are deployed uniformly within the simulation area and they move independently according to a random waypoint mobility model. The remaining 60 UEs of are uniformly distributed into up to eight clusters. The number of UEs in each cluster is also random. The UEs in the same cluster are located within a circle with a radius of 80 m. All UEs within one cluster follow the same cluster movement trajectory (defined by the center of the cluster) and each UE can move arbitrary within the cluster.

The channel between the FlyBS and the ground units (SBSs and UEs) is modeled as the air-to-ground (A2G) communication according to [20], with the suburban environment parameters ("suburban" channel model, i.e., $a = 4.88, b = 0.43, \eta_{LoS} = 0.1$ and $\eta_{NLoS} = 21$, see [20] for more details). The channel between the SBSs and the UE is modeled according to [21] with the path loss model $128.1+37.6log_{10}d$, where d (in km) is the distance between the UE and the SBS.

We consider 40 random realizations (deployments). In each realization, the positions of the UEs, corresponding trajectory of the FlyBSs, and the positions of the SBSs are random. The results of all realizations are then averaged out to suppress impact of randomness in the models. For the Q-learning training purposes, different settings of α and λ have been tested and we have observed that α= 0.8 and λ = 0.6 are the most suitable for the proposed algorithm. The values of CIO are determined from set {-6, -3, 0, 3, 6} dB [22]. Note that, we have tested also sets with smaller steps (1 and 2 dB), but there is no notable impact on performance. Hence, we select the larger step (3 dB), since the larger the step is, the smaller the Q-table is. Table I summarizes the major parameters used in our simulations.

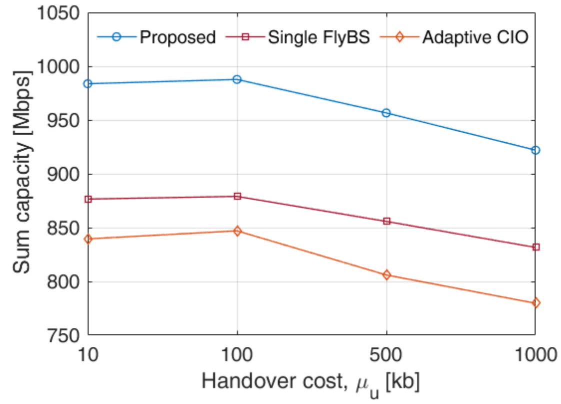


Fig. 2: Sum capacity of UEs served by four FlyBSs for different μ_u, c_{req}=25 Mbps.

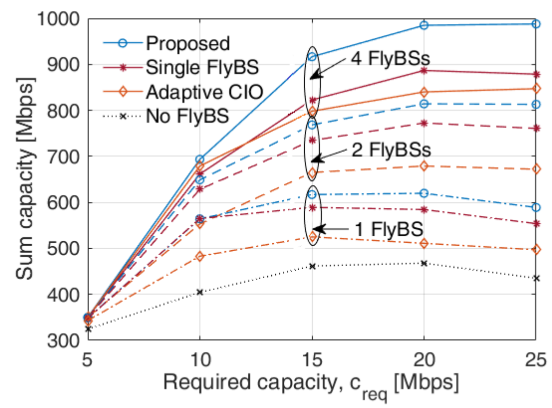


Fig. 3: Sum capacity of UEs served by FlyBSs.

The performance of the proposed Q-learning algorithm is compared with following benchmarks and state-of-the-art approaches: i) *no FlyBS* deployed, i.e., all UEs served only by the SBSs, as a benchmark to confirm that the deployment of the FlyBSs in our scenario is meaningful and the FlyBSs do not degrade the performance; ii) the CIO adjustment algorithm from [11], denoted as *Adaptive CIO*, which sets CIO according to the predefined relation between the value of CIO and the average load of the SBSs, iii) the reinforcement learning algorithm introduced in our prior work [16] for single FlyBS and without considering FlyBSs' interaction during handover process (denoted as *Single FlyBS* in figures). We consider the sum capacity of the UEs served by the FlyBSs, defined as $\sum_{n=1}^{N_u} c_n(t)$, as a performance indicator for the evaluation.

Before comparing our proposal with the competitive algorithms, let us demonstrate an impact of the handover cost μ_u on the sum capacity. Figure 2 shows the capacity of the UEs served by four FlyBSs achieved by various handover algorithms for different handover cost values. A low handover cost does not negatively impact the sum capacity, since the low handover overhead is easily compensated by an increase in the capacity of the handovering UE. However, for higher handover

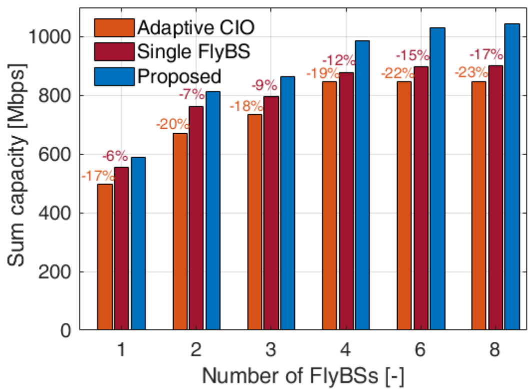
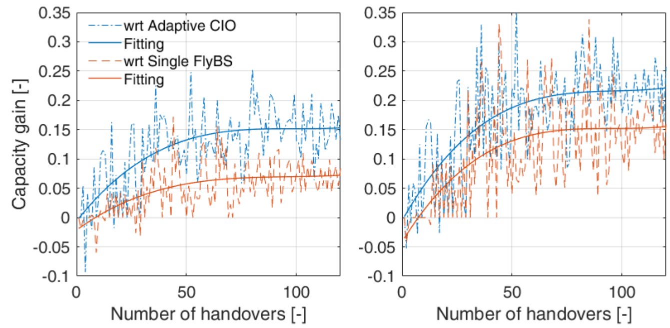


Fig. 4: Sum capacity of UEs served by FlyBSs, c_{req}=25 Mbps.

cost values, a decrease in the sum capacity is observed (with similar slope) for all algorithms. This decrease is because an improved sum capacity due to handover cannot longer compensate a high handover cost. Based on the handover management procedure defined by 3GPP [23], the handover overhead typically ranges in order of dozens to hundreds kb per UE. Hence, for further analysis, we select μ_u =100 kb.

Figure 3 shows the sum capacity of the UEs served by the FlyBSs. The sum capacity raises with c_{req} up to c_{req} = 20 Mbps. Then, for c_{req} higher than 20 Mbps, the sum capacity becomes almost constant or even starts slightly decreasing. The decrease in the sum capacity of the UEs with increasing c_{req} is because, while all UEs (served by any BS) require a higher capacity, the SBSs still have only the same bandwidth that can be allocated to the UEs. The proposed algorithm outperforms both the Adaptive CIO and the Single FlyBS algorithms and the gain increases with the number of FlyBSs. For 4 FlyBSs, the proposed algorithm increases the sum capacity by up to 19% and 12% compared to the Adaptive CIO and the Single FlyBS algorithms, respectively. This increase in the UEs' capacity is because the proposed algorithm prevents the SBSs' overloading and distributes the FlyBSs fairly among the SBSs by setting different CIO of the SBSs for each FlyBS.

More detailed view on the impact of the number of the FlyBSs on the sum capacity for c_{req}= 25 Mbps is provided in Figure 4. The proposed algorithm outperforms the Adaptive CIO and Single FlyBS algorithms for all numbers of the FlyBSs. Moreover, the relative capacity gain achieved by the proposed algorithm compared to the competitive schemes increases with the number of the FlyBSs. For 8 FlyBSs, the proposed algorithm outperforms the Adaptive CIO and Single FlyBS algorithms by 23% and 17%, respectively. The relative gain in capacity achieved by the proposed algorithm increases with the number of the FlyBSs, since the proposed algorithm prevents the FlyBSs from connecting to the same SBS simultaneously to avoid overloading of the SBSs. The relative capacity gain starts saturating for a higher number of the FlyBSs. This is due to the limited amount of bandwidth and the interference among the FlyBSs and the SBSs (the additional FlyBSs increase interference).



(a) 1 FlyBS (b) 8 FlyBSs

Fig. 5: Learning progress represented by gain in sum capacity with respect to *Adaptive CIO* and *Single FlyBS* algorithms over number of handovers performed by FlyBSs for c_{req}=25 Mbps.

Figure 5 illustrates the learning progress of the proposal after individual learning events, i.e., after each handover performed by the FlyBS. The figure depicts the gain achieved by the proposal in the sum capacity of UEs served by the FlyBSs with respect to the Adaptive CIO and Single FlyBS algorithms for 1 FlyBS (a) and 8 FlyBS (b). At the beginning of the learning, the gain of the proposed algorithm is rather small or even slightly negative in some steps compared to the Adaptive CIO and Single FlyBS algorithms. This is a result of the initial "random" learning when (almost) no information that would guide the selection of the CIO is available. However, after a short initial phase (roughly 30 handovers), the gain becomes always non-negative and increases with additional handovers. The capacity gain achieved by the proposed algorithm converges and becomes notably positive approximately after 60 handovers. The figure also illustrates fitting function for the gain with respect to the Adaptive CIO and Single FlyBS algorithms. The fitting function confirms that the proposed algorithm outperforms the Adaptive CIO and Single FlyBS algorithms in the sum capacity of UEs. Requiring only tens of handovers to learn the suitable values of CIO and to stabilize a highly positive gain in the sum capacity is sufficiently fast to deploy the proposed algorithm in real networks.

To demonstrate that the proposed algorithm does not have a negative impact on the UEs served by the SBSs, in figure 5, we show the sum capacity of the UEs served only by the SBSs. The UEs' capacity for all three compared algorithms is similar and the proposal even slightly increases the sum capacity of the UEs attached to the SBSs by up to 3.7% and 1.3% compared to the Adaptive CIO and the Single FlyBS algorithms, respectively. This confirms that the gain in the sum capacity of the UEs served by the FlyBSs introduced by the proposal is *not* at the cost of a degraded capacity of the UEs served by the SBSs. This conclusion is valid regardless of the number of FlyBSs in the network.

Figure 7 depicts the number of handovers performed by FlyBSs for c_{req}=25 Mbps. The proposed algorithm signifi-

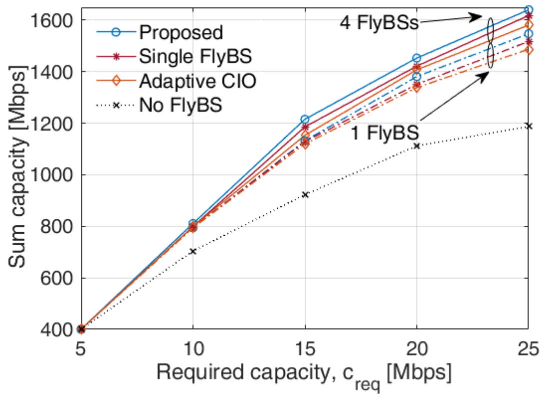


Fig. 6: Sum capacity of UEs served by SBSs for different number of FlyBS.

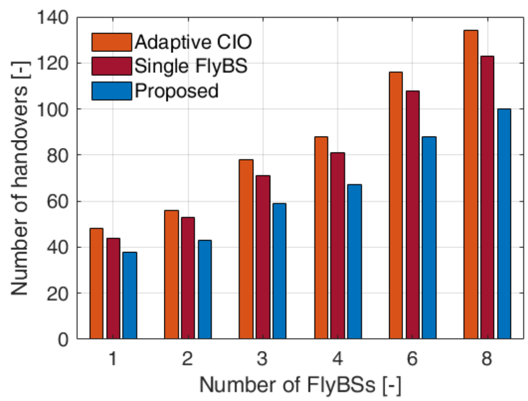


Fig. 7: Number of handovers performed by FlyBSs for c_{req}=25 Mbps.

cantly reduces the number of handovers performed by FlyBSs compared to the competitive schemes and, moreover, the reduction becomes more significant with increasing number of the FlyBSs. For example, if eight FlyBSs are deployed, the proposed algorithm outperforms the Adaptive CIO and Single FlyBS algorithms by 25% and 19%, respectively. The reduced number of handovers by the proposed algorithm is a result of preventing the SBS's overloading via consideration of the SBS load and taking the handover cost into account.

V. Conclusion

In this paper, we have proposed a novel algorithm managing handover of multiple FlyBSs among the SBSs to maximize the sum capacity of the UEs served by the FlyBSs while taking the handover cost into account. The proposed algorithm exploits Q-learning to adjust CIO of the SBSs for each FlyBS according to the load of SBSs and the traffic generated by the UEs served by the FlyBSs. The results show that the gain introduced by the proposed algorithm with respect to the competitive works increases with the number of FlyBSs deployed in the system. The proposed Q-learning based approach outperforms the state-of-the-art solutions in the sum capacity of UEs served by the FlyBSs and in the number of handovers performed by FlyBSs by up to 23% and 25%, respectively, if eight FlyBSs are deployed. Besides, the fast convergence of the proposed algorithm allows its practical application.

Future research should be focused on joint handover decision for the FlyBSs and UEs and the FlyBS positioning.

References

[1] Y. Zeng, R. Zhang and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, May 2016.

[2] B. Li, Z. Fei and Y. Zhang, "UAV communications for 5G and beyond: Recent advances and future trends," *IEEE Internet of Things Journal*, vol. 6, no. 2, April 2019.

[3] Y. Chen, W. Feng and G. Zheng, "Optimum placement of UAV as relays," *IEEE Communications Letters*, vol. 22, no. 2, Feb. 2018.

[4] S. Zeng, H. Zhang, K. Bian and L. Song, "UAV relaying: Power allocation and trajectory optimization using decode-and-forward protocol," *IEEE ICC workshops* 2018.

[5] M. Najla, Z. Becvar, P. Mach and D. Gesbert, "Positioning and Association Rules for Transparent Flying Relay Stations," *IEEE Wireless Communications Letters*, 2021.

[6] J. Angjo, I. Shayea, M. Ergen, H. Mohamad, A. Alhammadi and Y. I. Daradkeh, "Handover Management of Drones in Future Mobile Networks: 6G Technologies," *IEEE Access*, vol. 9, 2021

[7] 3GPP, "E-UTRA radio resource control (RRC) protocol specification (Release 8)," 3GPP, Tech. Rep. 36.331 V8.16.0, Dec. 2011.

[8] Z. Becvar, P. Mach, "Adaptive hysteresis margin for handover in femtocell networks", *ICWMC*, 2010.

[9] K. da Costa Silva, Z. Becvar, C. R. Francês, "Adaptive Hysteresis Margin Based on Fuzzy Logic for Handover in Mobile Networks with Dense Small Cells", *IEEE Access*, vol. 6, 2018.

[10] S. S. Mwanje and A. Mitschele-Thiel, "A Q-Learning strategy for LTE mobility Load Balancing," *IEEE PIMRC*, London, 2013.

[11] S. Su, T. Chih and S. Wu, "A novel handover process for mobility load balancing in LTE heterogeneous networks," *IEEE ICPS*, 2019.

[12] Q. Li, M. Ding, C. Ma, C. Liu, Z. Lin and Y. -C. Liang, "A Reinforcement Learning Based User Association Algorithm for UAV Networks," *International Telecommunication Networks and Applications Conference (ITNAC)*, 2018.

[13] M. M. U. Chowdhury, W. Saad and I. Güvenç, "Mobility Management for Cellular-Connected UAVs: A Learning-Based Approach," *IEEE ICC workshops*, 2020.

[14] W. Dong, X. Mao, R. Hou, X. Lv and H. Li, "An Enhanced Handover Scheme for Cellular-Connected UAVs," *2020 IEEE/CIC ICCC*, 2020

[15] Y. Chen, X. Lin, T. Khan and M. Mozaffari, "Efficient drone mobility support using reinforcement learning," *IEEE WCNC*, 2020.

[16] A. Madelkhanova, Z. Becvar, "Optimization of Cell Individual Offset for Handover of Flying Base Station," *2021 IEEE VTC - Spring*, 2021.

[17] Z. Becvar, M. Vondra, P. Mach, J. Plachy and D. Gesbert, "Performance of mobile networks with UAVs: Can flying base stations substitute ultra-dense small cells?," *European Wireless*, 2017.

[18] J. N. Laneman, D. N. C. Tse and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, Dec. 2004.

[19] R. Sutton "Reinforcement Learning: An Introduction" MIT Press, 1998.

[20] A. Al-Hourani, S. Kandeepan and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," *IEEE Wireless Communications Letters*, vol. 3, 2014.

[21] R. I. Bor-Yaliniz, A. El-Keyi and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," *2016 IEEE ICC*, 2016.

[22] K. Attiah et al., "Load Balancing in Cellular Networks: A Reinforcement Learning Approach," *2020 IEEE CCNC*, 2020.

[23] 3GPP, "Handover Procedures," 3GPP, TS 23.009 V16.0.0, Jul. 2020.