This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

Optimization of Cell Individual Offset for Handover of Flying Base Stations and Users

Aida Madelkhanova, *Student Member, IEEE,* Zdenek Becvar, *Senior Member, IEEE* and Thrasyvoulos Spyropoulos, *Member, IEEE*

***Abstract*—To ensure a seamless mobility of users in the scenario with flying base stations (FlyBSs) and static ground base stations (GBSs), an efficient handover mechanism is required. In this paper, we introduce new framework simultaneously managing cell individual offset (CIO) for handover of both FlyBSs and mobile users. Our objective is to maximize capacity of the mobile users while considering also a cost of handover to reflect potential excessive signaling and energy consumption due to redundant handovers. This problem is of a very high complexity for conventional optimization methods and optimal solution would require knowledge of information commonly not available to the mobile network. Hence, we adjust the CIO of FlyBSs and GBSs via reinforcement learning. First, we adopt Q-learning to solve the problem. Due to practical limitations implied by a large Q-table, we also propose Q-learning with approximated Q-table. Still, for larger networks, even the approximated Q-table can require a large storage and computation time. Therefore, we apply also actor-critic-based deep reinforcement learning. Simulation results demonstrate that all three proposed algorithms converge promptly and increase the communication capacity by dozens of percent while the handover failure ratio and the handover ping-pong ratio are reduced multiple times compared to state-of-the-art.**

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

I. Introduction

FLYING base stations (FlyBSs), essentially the unmanned aerial vehicles (UAVs) carrying a hardware for wireless communication, are seen as a suitable solution for the future mobile networks due to their flexible deployment and high mobility. The FlyBSs allow to extend the network coverage [1], [2] and/or to boost quality of service in a specific area [3], [4]. Due to a fast deployment, the FlyBSs are suitable also for emergency situations or short-time events [5]. However, an integration of the FlyBSs to the mobile networks introduces new challenges, such as finding optimal position of the FlyBS [6], optimizing the FlyBS's trajectory [7], or an association of user equipments (UEs) to the FlyBSs [8]. Besides, the problem of power allocation and trajectory optimization for the network with FlyBSs exploiting non-orthogonal multiple access (NOMA) is targeted, e.g., in [9] to improve the system security. In [10], the UE's association is optimized to increase the sum capacity in the scenario with the FlyBSs operating in mm-wave with massive multiple-input multiple-output antennas. Furthermore, in [11], the authors study the sum rate maximization in the network with FlyBSs and NOMA via optimization of the FlyBS's trajectory and precoding. In [12], intelligent reflecting surfaces (IRS) are employed to boost the network throughput and ensure a secure communication in the network with FlyBSs.

In addition to these works, another key challenge is to provide a seamless mobility of the FlyBSs among static ground base stations (GBSs) [13]. The trajectory of the FlyBSs serving mobile users is arbitrary and hard to be predicted, as it depends on a random 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 GBS providing connectivity of the FlyBS to the network. As a result, the mobile networks with FlyBSs can suffer from following major problems that we address in this paper: (a) rapid changes in quality of both the FlyBS-UE channels and the GBS-FlyBS channels may lead to unpredictable and frequent handovers and, consequently, to handover failures or packet losses and significant degradation in the Quality of Service (QoS); (b) unless the handovers of the FlyBSs among the GBSs are carefully and jointly coordinated with the handovers of the UEs among both GBSs and FlyBSs, some GBSs might end up severely overloaded through a single FlyBS handover.

In a common management of mobility of the UEs among the GBSs, the handover is typically initiated when a target GBS (i.e., the base station to which the handover should be performed) provides a channel of a higher quality than the current serving GBS. To avoid frequent handovers and/or handover failures, the decision on the handover is controlled and adjusted via parameters, such as a hysteresis, a time-to-trigger (TTT), or a cell individual offset (CIO) [14]. Thus, the handover is usually triggered when the target GBS provides the channel of a quality that is at least the hysteresis and/or the CIO above the quality of the channel to the serving GBS for an interval of the TTT. An increase in hysteresis, CIO, and/or TTT delays the handover triggering and, consequently, reduces the amount of handovers, especially those that are redundant (denoted often as ping-pong handover referring to a repeated switching between a pair of the GBSs). However, when the handover is over delayed, the quality of signal received by the UE degrades too much and the UE is not able

This work was supported by Ministry of Education, Youth and Sport of the Czech Republic under Grant No. LTT20004, and by Czech Technical University in Prague under Grants SGS20/169/OHK3/3T/13.

A. Madelkhanova and Z. Becvar are with the Faculty of Electrical Engineering, Czech Technical University in Prague, 166 27 Prague, Czech Republic (e-mail: aida.madelkhanova@fel.cvut.cz; zdenek.becvar@fel.cvut.cz).

T. Spyropoulos is with the Department of Communication Systems, EURECOM, Biot, France and with Electrical and Computer Engineering Dept., Technical University of Crete, Greece (e-mail: thrasyvoulos.spyropoulos@eurecom.fr).

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

to communicate with any GBS. Equally importantly, handover parameters (and, thus, also handover decisions) not only affect individual channel qualities, but also the total load of the serving GBS. For example, setting CIO of an overloaded GBS to a low value while setting a high CIO for neighboring GBSs allows to offload some UEs from the overloaded GBS to the neighboring GBSs. Adding the CIO into handover decision process can enhance the capacity of the UEs by triggering the handover of the UEs towards an underutilized GBS, which is able to provide a channel of maybe slightly worse quality, but much wider [15].

In the mobile networks comprising only GBSs (i.e., no FlyBS), the optimization of the handover decision parameters has been extensively addressed. For example, in [16], the authors propose an adaptation of the hysteresis according to relative qualities of the channels from the serving and neighboring GBSs. This approach reduces the number of handovers; however, it does not improve the UEs' throughput. In [17], the authors adapt the hysteresis via fuzzy logic to minimize the number of performed handovers. Nevertheless, an impact of the handover on the throughput of the UEs is not considered. A reactive load balancing algorithm based on Q-learning is developed in [18]. The reactive algorithm adapts the CIO of the serving and neighboring GBSs by a specific value so that the offset for the serving and neighboring GBSs is of the same absolute value, but opposite sign (e.g. –0.5 dB and +0.5 dB for the serving and neighboring GBSs, respectively). The CIO is adjusted according to a distribution of the cell-edge UEs in the area according to [18]. However, the UEs' distribution is typically unknown in practical scenarios. In [19], the authors adjust CIO for the load balancing purposes. Three predetermined thresholds are defined to distinguish four levels of the GBSs' load. A higher CIO is selected for the GBS with a lower load, and a lower CIO is set for the highly loaded GBSs. This CIO adjustment relieves the heavy traffic load of the GBS; however, it does not consider an impact of the handover on the throughput of UEs. In [20], the authors propose a machine learning-based framework to determine the optimal combination of CIO and hysteresis to maximize mean Signal to interference and noise ratio (SINR) of the UEs in the wireless network. The authors evaluate the performance of five different machine learning models for prediction of the mean SINR of the UEs with different combinations of CIO and hysteresis. Nonetheless, this technique requires a big and hard to collect set of training data to reach a sufficient accuracy. The problem of GBSs' CIO setting together with the transmission power optimization using deep reinforcement learning is considered in [15]. The authors propose an actor-critic-based framework, which adjusts the CIOs and the transmission power of GBSs. The proposed algorithm reflects the trade-off between the UEs' throughput and the number of covered UEs in the mobile network and improves the average throughput of the mobile network. However, an impact of the algorithm on the number of handovers and related signaling overhead is not considered.

Handover procedure is also challenging in the scenario with deployed mobile relays, mounted on trains or public transportation vehicles, since the mobile relays suffer from handover failures due to a high speed. In [21], the authors optimize the handover of mobile relays mounted on high speed trains. The authors rely on a predictability of the trains' movement on railways. The results show that the proposed scheme reduces mobile relays' handover time and signaling overhead. The performance of the handover of mobile relay installed at the roof-top of a bus is investigated in [22]. The proposed handover procedure reduces the overall power consumption and the number of performed handovers. The authors in [23] present a dual antenna handover scheme for the mobile relay represented by high speed trains. The proposal reduces both handover outage probability and communication interruption probability. Since the mobile relays are typically deployed on the trains of the public transportation vehicles, as expected in [21]–[23], the above-mentioned works assume the trajectory of mobile relays is predictable and the mobile relays are almost static with respect to served UEs. However, this is not valid in the scenarios with the UAVs acting as the FlyBSs serving the moving UEs, since the trajectory of the FlyBSs depends on the UEs' movement and do not follow a predictable pattern.

In contrast to a substantial research effort on handover in conventional mobile networks, only limited amount of works target handover management in the mobile networks with UAVs in general. Few works study the problem of handover in the mobile networks with the UAVs acting as the UEs (UAV-UE), i.e., not serving any ground UEs. The handover management for the UAV-UE via a dynamic adjustment of the GBSs' antenna tilt angles is outlined in [24]. The authors demonstrate that an intelligent antenna tilting reduces the number of handovers in a simple mobility scenario with the UAV-UE traveling along a linear trajectory. In [25], the authors propose a scheme adjusting the handover parameters for the UAV-UEs while the handover decision is based on the UAV-UE's trajectory to reduce the number of performed handovers. In [26], the handover based on the reinforcement learning is proposed to maximize the received signal quality at the UAV-UEs while minimizing the number of performed handovers. In [27], the authors propose a route-aware handover algorithm to improve a reliability of the UAVs' communication. This solution utilizes a flight path information to minimize a probability of the handover failure and to reduce the number of redundant handovers. Furthermore, the handover decision scheme for the UAV-UEs based on deep reinforcement learning is presented in [28] targeting to find a trade-off between the received signal strength and the frequency of handovers. Despite the encouraging results, the works [24] – [28] 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 a priori unknown and depends on the UEs' movement.

To our best knowledge, there is no paper dealing with the handover of the UEs among all BSs, i.e., both FlyBSs and GBSs, jointly with the handover of the FlyBSs among the GBSs while considering also the cost of handovers. If the UEs perform handovers among all BSs and, at the same time, the FlyBSs perform handovers among the GBSs, the

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

handovers occur frequently leading to excessive signaling overhead and related additional energy consumption, which can be critical for the energy constrained FlyBSs. Furthermore, since not only the UEs, but also the FlyBSs move in an unpredictable way, the overall scenario is significantly more dynamic and less predictable compared to the scenario with only GBSs or with the UAV-UEs. Both conventional and state-of-the-art handover algorithms, however, do not take such dynamicity into account and can lead to either redundant handovers and/or to overloading of the BSs resulting into frequent handover failure. In this paper, we focus on these critical and challenging aspects of joint handovers of the UEs among all BSs (comprising both FlyBSs and GBSs) and handovers of the FlyBSs among the GBSs and we propose a novel handover framework maximizing the sum capacity of the users while avoiding redundant handovers and handover failures. The major contributions of our paper are summarized as follows:

- We propose a framework for setting of the CIO of individual BSs (including both GBSs as well as FlyBSs) to increase the sum capacity of the UEs while avoiding redundant handovers. Due to dynamic nature and unpredictable behavior of the UEs (and consequently also FlyBSs serving these UEs) together with an indirect and unpredictable relation between currently taken decision and its future impact, we employ reinforcement learning, which provides a solution for the problems with an unknown environment assumed in our case.
  First, we consider a tabular Q-learning framework for a dynamic CIO adjustment. The Q-learning is known to provably converge to the optimal solution and, hence, provides an efficient mobility support in the sky.
- Despite theoretical advantages of the tabular Q-learning, prohibitively large computation and storage resources are required by the Q-learning. Therefore, we also consider an approximate Q-learning that greatly reduces size of the Q-table resulting in a lower computation and storage requirements. The approximate Q-learning leads to only a negligible decrease (below 2%) in the UEs' capacity. However, the number of ping-pong handovers is slightly increased compared to the original Q-learning. Still, even the approximate Q-learning can impose notable computation and storage resource requirement in large mobile networks.
- To avoid the increased number of ping-pong handovers due to approximate Q-learning and to overcome the problem of computing and storage resources in the large mobile networks, we extend our work towards the actor-critic deep reinforcement learning framework for dynamic CIO setting. Simulations demonstrate that the actor-critic approach enables fast-convergence with only a marginal degradation in the UEs' capacity while no negative impact on handover failure and ping-pong effect is observed compared to the tabular Q-learning with complete Q-table.
- We demonstrate that all three proposed solutions lead to a notable increase in the UEs' capacity and, at the same time, to a significant decrease in the number of handover failures and ping-pong handovers compared to state-of-the-art works.

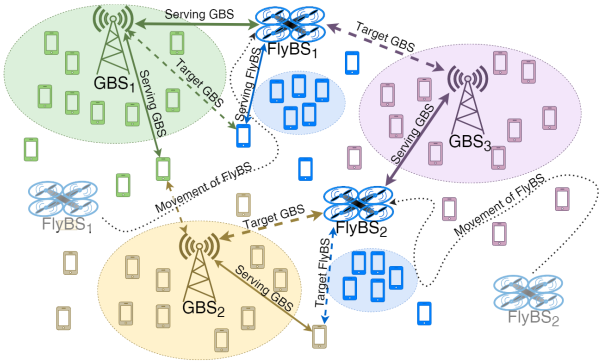


Fig. 1. System model with multiple GBSs and multiple FlyBSs serving mobile UEs. The FlyBSs serve the UEs and relay their data to the GBSs; while the GBSs can server directly both the UEs as well as the FlyBSs.

This paper is an extension of our prior works [29], [30], where the handover of only FlyBS(s) is considered and the handover of the UEs connected to GBSs is not taken into account.

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 control of the CIO for the GBSs. The simulation results and their discussion are provided in Section IV. Last, Section V concludes the paper.

II. System model

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

A. System model

In this subsection, first, the communication network model is presented. Afterwards, the channel models are defined and the handover procedure for both the UEs and the FlyBSs is described.

1) Network model: We assume N UEs deployed in an arbitrary area covered with K_G conventional GBSs and additional K_F FlyBSs. Hence, in total, the area is covered by $K = K_G + K_F$ BSs. Note that the label BS represents jointly the GBSs and the FlyBSs in this paper. Each UE in the system is assumed to require the communication capacity c_{req} from the BS it is connected to. The required capacity c_{req} can be possibly a different for each UE. Out of N UEs, N_f UEs are connected to the mobile network via the FlyBSs, which relay the communication between the GBS and these UEs.

The position of the n-th UE changes over time and the FlyBSs moves according to the movement of the connected UEs. Without loss of generality, the position of each FlyBS corresponds to the center of gravity of all UEs associated to this FlyBS, as suggested in [31]. Note that the principle of the proposed solution for the CIO adjustment does not depend on the specific positioning of the FlyBSs and can be applied

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

on the top of any other approaches. To maintain a reliable connectivity, each FlyBS performs handovers during flight and, thus, the association of the FlyBSs to the GBSs changes over time. We define a binary parameter $\beta_{g,f}$ indicating if the f-th FlyBS is associated to the g-th GBS ($\beta_{g,f}$= 1) or not ($\beta_{g,f}$= 0).

2) Channel model: We consider the downlink communication from the GBSs to the UEs either directly or via the FlyBSs. The SINR $\gamma_{g,n}$ observed by the n-th UE served directly by the g-th GBS is defined as:

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

where P_g is the transmission power of the g-th GBS serving the n-th UE, $h_{g,n}$ is the channel gain between the n-th UE and the g-th GBS, the term $\sum_{i=1,i\neq g}^{K} P_i h_{i,n}$ 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.

The SINR $\gamma_{f,n}$ observed by the n-th UE from the f-th FlyBS is defined as:

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

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.

Last, the SINR at the f-th FlyBS receiving data from the g-th serving GBS is expressed as:

$$\gamma_{g,f} = \frac{P_g h_{g,f}}{\sum_{i=1,i\neq g}^{K} P_i h_{i,f} + \sigma^2} \tag{3}$$

where $h_{g,f}$ is the channel gain between the g-th serving BS and the f-th FlyBS, $\sum_{i=1,i\neq g}^{K} P_i h_{i,f}$ 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.

We adopt decode and forward relaying, hence, the relaying channel capacity for the communication of the the n-th UE via the f-th FlyBS is, in line with [32], defined as:

$$c_n = \frac{B_n}{2} min\{log_2(1+\gamma_{g,f}), log_2(1+\gamma_{f,n})\} \tag{4}$$

where B_n denotes the bandwidth of the n-th UE's channel expressed as the bandwidth requested by the UE to meet c_{req} ($c_n = c_{req}$):

$$B_n = \frac{c_{req}}{log_2(1+\gamma_{k,n})} \tag{5}$$

where $\gamma_{k,n}$ is the SINR observed by the n-th UE served by the k-th BS. The bandwidth allocation is not directly related to the handover decision itself. Thus, we assume the bandwidth is allocated according to the UEs' SINR in descending order (i.e., the UE with the highest SINR is allocated first). If the BS has enough resources to meet c_{req} required by the UE, B_n is allocated to this UE. This is repeated for next UEs until there is not enough remaining bandwidth that can satisfy requirements of any further UE. The remaining bandwidth of the BS is, then, divided equally among the rest of UEs served by the given BS.

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

$$\rho_k = \frac{\sum_{n\in N} \beta_{k,n} B_n + \sum_{f\in K_f} \left(\beta_{k,f} \sum_{n\in N} \beta_{f,n} B_n\right)}{B} \tag{6}$$

where the binary parameter $\beta_{k,n} \in \{0,1\}$ indicates if the n-th UE is associated to the k-th BS ($\beta_{k,n}$ = 1) or not ($\beta_{k,n}$ = 0), the term $\sum_{n\in N} \beta_{k,n} B_n$ represents the sum bandwidth allocated to the UEs directly connected to the k-th BS, and the term $\sum_{f\in K_f} \left(\beta_{k,f} \sum_{n\in N} \beta_{f,n} B_n\right)$ represents the amount of bandwidth allocated to the f-th FlyBS if the f-th FlyBS is associated to the k-th BS to serve the UEs connected via the f-th FlyBS.

3) Handover procedure: Handover between the serving BS and the target BS is triggered according to commonly adopted event A3, defined by 3GPP (see, e.g., [14]). Hence, the UE performs handover to the target BS if the following equation is satisfied for at least the period of TTT:

$$P_t h_{t,n} + CIO_t - Hys > P_s h_{s,n} + CIO_s \tag{7}$$

where the indices s and t correspond to the parameters of the serving and target BSs, respectively, and Hys is the value of the hysteresis in dB. The channel quality is represented by the received signal strength expressed as $P_k h_{k,n}$, where $h_{k,n}$ is the channel gain between the k-th BS and the n-th UE.

Each FlyBS can perform handover(s) among the GBS during the service provisioning to the UEs. Like the common UEs in the mobile networks, also the FlyBS measures the channel quality from the neighboring GBSs. The channel quality measurement report is periodically sent to the serving GBS in a similar way as the common UEs report their channel quality in the mobile networks [24]. Based on the measurement results, the handover of the FlyBS to one of the neighboring GBSs is triggered if the condition in (7) is satisfied for at least the period of TTT.

B. Problem formulation

Our objective is to optimize handover decision in the mobile networks encompassing both GBSs as well as FlyBSs serving mobile UEs via optimization of CIO. The CIO setting directly impacts the handover decision as indicated in (7). Handovers resulting from the chosen CIO may increase the capacity of both FlyBSs as well as UEs by handovering to the less congested BS. Thus, our objective is to adjust the CIOs of all BSs (i.e., both GBS as well as FlyBSs) 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 the energy consumption in both the communication network and the handovering devices (UE or FlyBS) [33]. Thus, the number of handovers should be

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

accounted for to enable affordable cost for network operation. Hence, the targeted problem is formulated as:

$$\boldsymbol{CIO^*} = \underset{\boldsymbol{CIO} \in O}{\operatorname{argmax}} \sum_{n=1}^{N_f} c_n - \mu \tag{8}$$

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. In this work, we assume that each UE is associated to just one BS and each FlyBS is associated to just one GBS.

III. Proposed adaptive CIO adjustment based on reinforcement learning

Our objective is to optimize the handover decision at a current step and, consequently, improve the performance in all subsequent steps. Such problem is complex and non-trivial due to a high randomness of the mobile network environment caused by the mobility of both the UEs and the FlyBSs. Moreover, the FlyBSs change their positions as the served UEs move, hence, the positions of the FlyBSs are hard to predict and do not follow any easily predictable pattern. Since we do not know future movements of the UEs and the FlyBSs, the decision on a change in CIO influences the future performance in an indirect and unpredictable way. Thus, the decisions should be only taken based on currently known information.

To solve the problem defined in (8) via conventional optimization techniques, a very accurate modeling of the optimized system would be required. Nevertheless, the modeling of the optimized system is not possible due to a significant uncertainty in the behavior of UEs and FlyBSs and due to the indirect and unpredictable impact of changes in CIO on future performance. Besides, our targeted problem falls under NP-hard problems, because of the non-linear and indirect coupling of two main variables characterizing our problem, i.e., between the CIO and the capacity of UEs. In addition, the complexity of such problem increases exponentially as the network expands (in terms of the number of BSs).

Thus, we apply reinforcement learning to solve our problem of the CIO setting. The adopted model-free reinforcement learning-based solution allows to solve the defined problem without requiring explicit system modeling, which is not available for our problem. Furthermore, the dynamic nature of reinforcement learning with the ability to adapt real-time observations is also suitable for the dynamic network with moving FlyBSs and UEs.

Unlike other machine learning algorithms requiring to generate data for an offline training phase, reinforcement learning allows the mobile network to learn and improve its decision by interacting with an unknown environment and exploiting received feedback. Reinforcement learning is a sequential decision making control algorithm, which can learn to optimally tradeoff the decision impact of the immediate step with the impact of future decisions for forthcoming system states, as required to solve our problem.

We propose the reinforcement learning-based algorithm to obtain the optimal CIO adjustment policy for the serving as well as target BSs. In following subsections, we first present the proposed Q-learning based CIO adjustment scheme. Then, we introduce the extended Q-learning based solution with approximate Q-table to relax requirements related to the Q-table size in practical applications. Afterwards, we present an actor-critic-based approach, which allows to completely circumvent the problems with the Q-table size even in large mobile networks while mitigating a small performance drop introduced by the approximate Q-table.

A. Q-learning based CIO adjustment

The 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 if the particular action from the set A is taken, and R is the reward function [35]. At each state, the MDP takes the action maximizing the expected sum of discounted future rewards. To solve the MDP corresponding to reinforcement learning, we use Q-learning. In Q-learning, using the iterative process, the agent learns the action-value function $Q(s_t, a_t)$, which indicates how good the action a_t performed in the state s_t is. The learned value $Q(s_t, a_t)$ is updated as follows:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left[\, r_t + \lambda(maxQ(s_{t+1},a)) - Q(s_t,a_t)\right], \tag{9}$$

where r_t is the immediate received reward, s_{t+1} is the next state, $\alpha \in \langle 0, 1\rangle$, represents the learning rate balancing new information against previous knowledge, and $\lambda \in \langle 0, 1\rangle$ is the discount factor balancing between the immediate and future rewards.

Now, we define the reward function and the sets of states and actions for our targeted problem.

Reward function: The reward function $r(t)$ should mimic the objective of the formulated problem. Hence, considering the problem formulation, the reward function reflects the objective to maximizing the sum capacity of UEs served by FlyBS and avoiding the redundant handovers by taking into account the cost associated to handover events. Thus, we define the reward function as:

$$r(t) = \frac{\sum_{n=1}^{N_f} c_n(t)}{N_f c_{req}} - \left(\sum_{i=1}^{n_h} \rho_i \frac{\rho_{t,i}}{\rho_{s,i}} + n_h \mu_u \right), \tag{10}$$

where $\rho_{t,i}$ and $\rho_{s,i}$ correspond to the load of target and serving BSs, respectively, ρ_i corresponds to the load implied by the UE performing handover, n_h is the number of UEs performing handover at the same time slot (for handover of the FlyBS, n_h is equal to the number of UEs served by this FlyBS), and μ_u denotes the handover cost for the UE. The term $\sum_{i=1}^{n_h} \rho_i \frac{\rho_{t,i}}{\rho_{s,i}}$ entails preventing handovers from the underloaded BSs to the overloaded BSs to avoid unnecessary handovers and handover failures caused by an overloading of the BSs.

State Space: In this work, the state comprises of two elements: 1) the load of the BSs; and 2) the load implied to the BS by the UE(s) performing handover. Thus, the set of states $S(t)$ is defined as a vector $S(t) = [\rho_1(t), \rho_2(t), \ldots, \rho_k(t), ..., \rho_K(t), \rho_h(t)]$ of a length $K + 1$,

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

where $\rho_k(t) \in \{0 : 0.01 : 1\}$ corresponds to the load of the k-th BS ($k \in \langle 1, K\rangle$) at the time t, $\rho_h(t) \in \{0 : 0.01 : 1\}$ is the load (i.e., resources required to meet c_{req}) implied by the UE or by the FlyBS performing handover at the time t. For the purposes of the state definition, the load values are rounded to two decimal points in order to discretise the state space with a negligible loss in accuracy (maximum inaccuracy is ±0.5%). Hence, the size of the states' space is defined as $(0 : 0.01 : 1)^{K+1}$.

Action Space: The agent controls the CIO of all BSs via the actions $A(t)$. The action is understood as a selection of the CIO for each BS. We present the action space at the time t as a vector $A(t) = [CIO_1(t), CIO_2(t), ..., CIO_k \dots, CIO_K(t)]$ of a length K, where the $CIO_k(t)$ corresponds to the CIO of the k-th BS ($k \in \langle 1, K\rangle$) 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 (i.e., with L possible values of CIO), where CIO_{min} and CIO_{max} are the minimum and maximum possible CIOs in the system, respectively. This creates the action space of L^K possible actions.

This definition of the state and action spaces for Q-learning is further referred to as *Proposal QL* in the rest of the paper.

In the Q-learning, the agent determines when and how much to explore the state-action space before exploiting the learned knowledge. To balance exploration and exploitation, we adopt the ϵ-greedy policy, where the agent tries to obtain the highest reward at each training step, however, the agent also checks for other actions to discover those that can potentially improve the estimated future reward [35]. The learning starts with ϵ = 1 to explore large space of possible actions and to avoid local optima. Then, ϵ is continuously reduced to ϵ = 0 by multiplying ϵ with a decay factor at each learning step [24].

B. Q-learning based CIO adjustment with approximate Q-table

In our setup, the state and action spaces grow exponentially with the number of BSs. Consequently, the Q-table of the *Proposal QL* becomes large and can be difficult to train. Therefore, we further propose to reduce the dimensions of the Q-table by approximation of the spate and action space. This approach with the approximate Q-table, which has the reduced state and action spaces, is further referred to as *Proposal AQL*. We define reward suction and state and action spaces as follows.

Reward function: The reward function for the *Proposal AQL* is defined in the same way as for the *Proposal QL*, see (10).

State Space: To reduce the state space, we define *M* predetermined thresholds (THs) to divide the load of the BSs into *M+1* levels:

$$s_k(t) \leftarrow \begin{cases} 1, & \text{if } \rho_k(t) < TH_1 \\ 2, & \text{if } TH_1 \leqslant \rho_k(t) < TH_2 \\ & \vdots \\ M, & \text{if } TH_{M-1} \leqslant \rho_k(t) < TH_M \\ M+1, & \text{otherwise,} \end{cases} \tag{11}$$

where $\rho_k(t) \in [0, 1]$ corresponds to the load of the k-th BS ($k \in \langle 1, K\rangle$) at the time t.

The proper setting of the thresholds is crucial for the performance. We use a standard state aggregation technique [35] and define state elements as the load levels of the BSs, each level corresponds to specific and unique range of the load. Thus, the state $S'(t)$ for the Q-learning based CIO adjustment with approximate Q-table is defined as a vector $S'(t) = [s_s(t), s_1(t), s_2(t), \dots, s_k(t), \dots, s_K(t), s_h(t)]$ of a length $K+1$, where $s_s(t) \in \langle 1, M+1\rangle$ corresponds to the load level of serving BS at the time t, $s_k(t) \in \langle 1, M+1\rangle$ is the load level of the k-th neighboring BS (for $k \in \langle 1, K-1\rangle$, and $s_h(t) \in \langle 1, M+1\rangle$ is the load level implied by the UE or the FlyBS performing handover at the time t. The size of the state space for the *Proposal AQL* is $(1, M+1)^{K+1}$.

Action Space: To reduce also the action space, we introduce relative CIO values so that (7) is rewritten as:

$$P_t h_{t,n} + CIO_{s\to t} > P_s h_{s,n} + Hys \tag{12}$$

where $CIO_{s\to t} = CIO_t - CIO_s$ is the relative CIO value of the s-th serving BS with respect to the t-th target BS. As a result, each serving BS has a single CIO value for each neighbor BS. Hence, the action is understood as a selection of the relative CIO values of the serving BS with respect to its all neighboring BSs: $A'(t) = [CIO_{s\to 2}(t), CIO_{s\to 3}(t), \dots, CIO_{s\to k}(t), ..., CIO_{s\to K}(t)]$, where $CIO_{s\to k}(t)$ corresponds to the relative CIO of the serving BS s with respect to its k-th neighboring BS ($k \in \langle 2, K\rangle$). In comparison with the action set for the *Proposal QL*, where the CIO is assigned to each individual BS, here, the CIO is assigned to a pair of the serving and neighboring BSs. Consequently, the number of possible actions is only L^{K-1}.

C. CIO adjustment based on Actor-Critic deep reinforcement learning

The Q-learning uses the Q-table to store value functions of each state action pair for iterative computations. Even for the approximate Q-table in the *Proposal AQL* introduced in Section III-B, the number of states and actions increases exponentially with the number of BSs and it is impractical and expensive to compute and store all value functions for every state action pair within the Q-table for large communication networks. Moreover, the state quantization performed in the *Proposed AQL* in Section III-B introduces quantization noise, which might impede the algorithm to find the true optimal policy. Thus, we further replace Q-learning with deep reinforcement learning, which uses the neural network to replace the role of Q-table. More specifically, we use the actor-critic algorithm as the deep reinforcement learning framework, since the actor-critic provides fast convergence properties and a capability to deal with a large action space [36], as in our optimization problem. The actor-critic is a deep reinforcement learning framework splitting the model into two components, an actor and a critic, to combine benefits of both the actor-only (e.g., natural gradient [37]) and the critic-only (e.g., Q-learning [38], SARSA [39]) approaches. In the actor-critic framework,

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

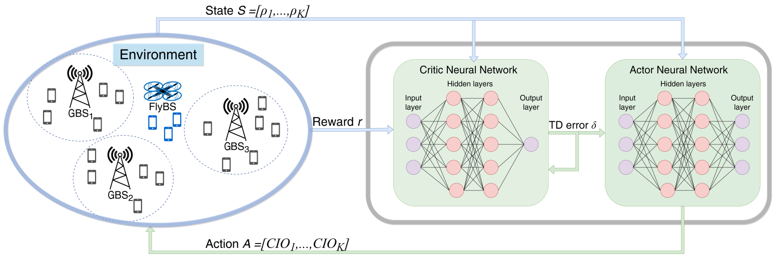


Fig. 2. Structure of the actor-critic deep reinforcement learning agent.

the learning process alternates between the policy evaluation by the critic and the policy improvement by the actor.

The structure of the actor-critic agent proposed to solve our problem is shown in Figure 2. In the proposed scheme, the actor-critic algorithm makes decisions on CIO setting in following way. First, the actor neural network define parameterized policy and chooses the CIO for all BSs according to the observed load status of the BSs. Then, the critic neural network evaluates the current policy by processing the rewards received from the environment and calculates the Temporal Difference (TD) error. Both the critic and actor neural networks are updated based on the TD error, which represents the error between the estimated value and the true value of the state-value function.

The TD error is commonly used to update the evaluated state-value function $V(s_t)$ in the critic to enhance the learning efficiency [40] and to train the critic neural network. The TD error $\delta(s_t)$ is expressed as:

$$\delta(s_t) = r_t + V_\omega(s_{t+1}) - V_\omega(s_t) \tag{13}$$

where $V_\omega(s_t)$ is the state-value function parameterized by the vector ω.

The critic updates the weights of the neural network according to the square of TD error $\delta^2(s_t)$:

$$\omega_{t+1} \leftarrow \omega_t + \alpha_c \nabla_\omega \delta^2(s_t) \tag{14}$$

where α_c is the learning rate of the critic.

The aim of the actor is to take the CIO adjustment decision maximizing the expected cumulative rewards according to the current state. The weight θ of the actor neural network is updated using TD error and the policy gradient as:

$$\theta_{t+1} \leftarrow \theta_t + \alpha_a \nabla_\omega [log \pi_\theta(a_t|s_t)] \delta(s_t) \tag{15}$$

where α_a is the learning rate of the actor, and $\pi_\theta(a_t|s_t)$ is the output probability for each action calculated by the actor.

To apply the actor-critic algorithm for the dynamic CIO setting, we interpret the optimization problem as the MDP. We use the definition of reward, states, and actions as in Section III-A. The state comprises of the load of the BSs and the load (i.e., required resources to meet c_{req}) implied by the UE or by the FlyBS performing handover at the time t, hence, the state space is defined as $S(t) = [\,\rho_1(t), \rho_2(t), \ldots, \rho_k(t), ..., \rho_K(t), \rho_h(t)]$. The action is understood as a selection of the CIO for each BS and the action space is defined as $A(t) = [CIO_1(t), CIO_2(t), ..., CIO_k ..., CIO_K(t)]$. The reward for the actor-critic approach is defined in the same way as for the *Proposal QL*, see (10). Note that concrete hyperparameters of both actor and critic neural networks are specified and explained in Section IV-A.

IV. Performance evaluation

In this section, models, scenario, and deployments used for performance evaluations are outlined. Afterwards, the performance metrics and the competitive state-of-the-art algorithms are defined. Last, the results of simulations are presented and performance of the proposal is compared with the state-of-the-art algorithms.

A. *Simulation models and scenarios*

The simulations are performed in MATLAB. We consider a suburban scenario with the simulation area of 1000×1000 m. Within this area, three conventional GBSs are deployed randomly with a minimum inter-site distance of 500 m. Furthermore, up to six 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 [31]. Note that the proposed solution is suitable for any other approach of the FlyBSs' positioning and we adopt center of gravity for its low complexity. The BSs serve 150 UEs moving with a random speed varying between 1 and 3 m/s and all UEs are active all the time. Out of all UEs, 60 UEs are randomly distributed and deployed uniformly around GBSs within a circular area with a radius of 150 m and these UEs move arbitrary within this circular area. Another 30 UEs are deployed uniformly within the whole simulation area and move independently according to a random waypoint mobility model [41]. The remaining 60 UEs follow the cluster movement model according to [42], [43]. These 60 UEs are, thus, uniformly distributed in up to six 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). The cluster movement within the simulation area is inline with a random waypoint mobility model [41]. A movement of each UE within the cluster is arbitrary in the whole simulation area. We assume the same c_{req} of all UEs for a clarity of the following explanations and results' presentation. However, our proposed solution is suitable for any, even diverse c_{req} for individual UEs.

Handover procedure is triggered according to A3 event (see (7)). We set hysteresis Hys and TTT to 3 dB and 0.16 s, respectively in line with [44].

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Simulation area	1000 × 1000 m
Carrier frequency	2 GHz
Tx power of GBS/FlyBS	23/15 dBm
Bandwidth of GBS	100 MHz
GBS/FlyBS/UE height	30/80/1.5 m
Number of UEs	150
Hysteresis margin	3 dB
TTT	0.16 s
Time step	1 s
CIO set	{-6, -3, 0, 3, 6} dB [47]

The channel between the FlyBS and any ground unit, i.e., GBSs or UEs, is modeled as the air-to-ground (A2G) communication according to [45], 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 [45] for more details). The channel between the GBSs and the UE is modeled according to [46] with the path loss model $128.1 + 37.6 log_{10} d$, where d (in km) is the distance between the UE and the GBS.

We consider 100 random and independent realizations with 25.000 time steps per realization and with a duration of each step of 1 second. In each realization, the positions of the UEs, corresponding trajectory of the FlyBSs, and the positions of the GBSs are random. The results of all realizations are then averaged out to suppress an impact of the randomness in the models.

For the Q-learning training purpose, different settings of α and λ have been tested and we have observed that $\alpha = 0.5$ and $\lambda = 0.6$ are the most suitable for the proposed algorithm. The values of CIO are determined from the set $\{-6, -3, 0, 3, 6\}$ dB [47]. Note that, also in this case, we have tested other sets with smaller steps of 1 and 2 dB, but there is no notable impact on the performance. Hence, we select the step of 3 dB, since the larger the step is, the smaller the Q-table is. Table I summarizes the major parameters used in our simulations.

The actor-critic agent consists of two fully connected neural networks as the approximators for the actor and the critic. The actor neural network has four hidden layers, each with 120 neurons and with ReLU adopted as the activation function. The output layer of the actor has L^K neurons. Since the action space is discrete, we use softmax function at the output layer of the actor neural network to obtain the scores of each action. The critic neural network, which computes the value of the chosen action, has three hidden layers of 120 neurons in each layer and with ReLU again used as the activation function. The output layer of the critic has one neuron. The number of hidden layers and the number of neurons in each hidden layer are set by a trial and error approach. Since the critic evaluates the decision made by the actor, the learning rate of the actor neural network should be smaller than that of the critic neural network to make the actor converging slower than the critic. We set the learning rate of the actor and critic neural networks experimentally via a trial and error approach to 0.01 and 0.001, respectively.

B. Performance metrics

Three metrics commonly adopted for an evaluation of the handovers in mobile networks are considered for the performance evaluation: capacity of UEs, handover failure ratio, and handover ping-pong ratio. We define these metrics as follows.

The *capacity* of UEs is understood as the summation of the communication capacities of all UEs averaged out over the simulation period T, i.e., the capacity is defined as $\frac{1}{T} \sum_{t=1}^{T} \sum_{n=1}^{N} c_n$.

A *handover failure* occurs when the UE fails to complete the handover procedure after the handover is triggered. In our case, the handover failure is determined according to the downlink SINR. Hence, when SINR is lower than the threshold Q_{out} (set to –8dB in our simulations according to [49]), a bad channel condition is indicated and the timer T310 is triggered. The handover failure is declared when T310 expires. We set the timer T310 to 1s, corresponding to a default value in 3GPP standards for 5G [14]. We measure the performance via *handover failure ratio (HFR)* defined as the ratio between the number of handover failures N_{fail} and the number of handovers (given by the sum of the number of the failed handovers N_{fail} and the number of successful handovers N_{suc}):

$$HFR = \frac{N_{fail}}{N_{fail} + N_{suc}} \tag{16}$$

The *ping-pong handover* is the frequent handover from one BS to another and back in a short time. The more this phenomenon occurs, the more handovers are processed and more signaling messages are generated. For this reason, the ping-pong effect should be avoided. The *handover ping pong ratio HPR* is defined as follows. If a connection is handed over to a new BS and handed back to the original serving BS in less than a critical time, denoted as minimum time-of-stay (t_{MTS}), the handover is considered as the ping pong handover [49]. The *handover ping pong ratio* represents the number of ping pong handovers N_{PP} divided by the total number of successful handovers N_{suc} (excluding the failed handovers), i.e.:

$$HPR = \frac{N_{PP}}{N_{suc}} \tag{17}$$

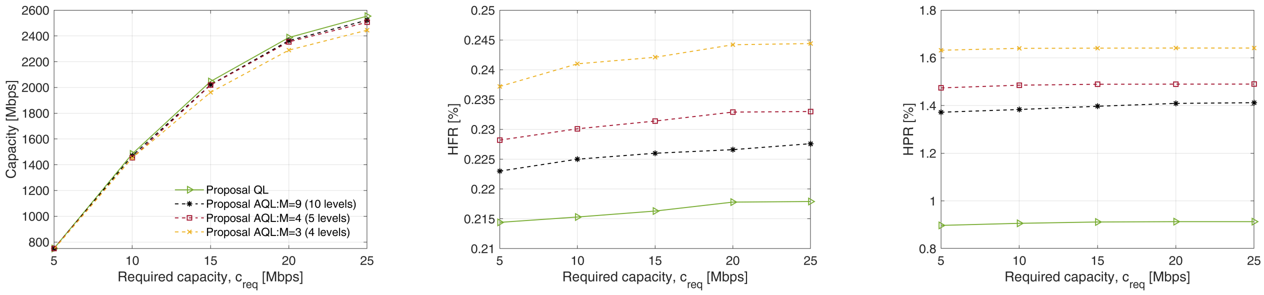
C. Competitive algorithms

The proposed algorithms are compared with following benchmarks and recent state-of-the-art works to demonstrate superiority of our proposal:

1) *no FlyBS*, i.e., all UEs are served only by the GBSs with CIO set to 0 dB for all GBSs [48]; this benchmark serves to confirm that the deployment of the FlyBSs in our scenario is meaningful and the FlyBSs do not degrade the performance;
2) *Adaptive CIO* adjustment algorithm from [19], which sets CIO based on predetermined GBSs' load thresholds and targets to minimize the number of performed handovers;
3) *Only FlyBSs CIO* algorithm, introduced in our prior work [30], adopting the Q-learning for the handover of FlyBSs, however, taking only the handover of FlyBSs

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342



(a) (b) (c)

Fig. 3. Impact of the number of thresholds (M) for Q-table reduction on: (a) the capacity of all UEs; (b) handover failure ratio HFR for all handovers; (c) handover ping-pong ratio HPR for all handovers.

into account while considering neither the UEs' handover nor the handover cost;

4) *Exhaustive search* checks all possible CIO settings for the handover of each UE and FlyBS and picks the CIO yielding the highest UEs' capacity in every time step to determine the maximum achievable capacity.

D. Simulation results

In this subsection, we first demonstrate the impact of the state space reduction on the performance of the *Proposal QL* and the *Proposal AQL*. Then, we show an impact of the handover cost on capacity achieved by all algorithms. Afterwords, we evaluate the capacity achieved by the algorithms for different numbers of deployed FlyBSs and we demonstrate the learning progress of the proposed algorithms. Last, we show the handover failure and ping-pong ratios reached by individual algorithms.

1) Impact of Q-learning table size reduction: Before comparing our proposal with the competitive state-of-the-art algorithms, let us demonstrate an impact of the Q-table size and its reduction (i.e., impact of the threshold for the BS load levels M) on the performance of the *Proposal QL* and the *Proposal AQL*. Individual subplots in Figure 3 show the capacity of the all UEs (subplot a), the handover failure ratio HFR for all handovers (subplot b), and the handover ping-pong ratio HPR for all handovers (subplot c). We investigate performance of the *Proposal AQL* for M equal to nine, four, and three BS's load thresholds corresponding to ten, five, and four load levels, respectively. The difference in the capacity achieved by the *Proposal AQL* with $M = 9$ and 4 predetermined thresholds is negligible (less than 0.5%) while a further reduction to $M = 3$ decreases the capacity by 5%. At the same time, the *Proposal AQL* with M equal to 9 and 4 reaches the capacity only 1.5% and 2% below the *Proposal QL*, respectively.

Figure 3b shows the Q-table approximation has a marginal impact on the HFR and the *Proposal AQL* adds only less than 0.01% (for M=9) and 0.03% (for M=3) to the HFR compared to the *Proposal QL* with full-size Q-table. Furthermore, Figure 3c demonstrates the approximation of the Q-table in the *Proposal AQL* leads to an increase in the HPR from roughly 0.9% reached by the full Q-table in the *Proposal QL* to about 1.4% and 1.65% in case of the *Proposal AQL%* with M=9 and M=3, respectively.

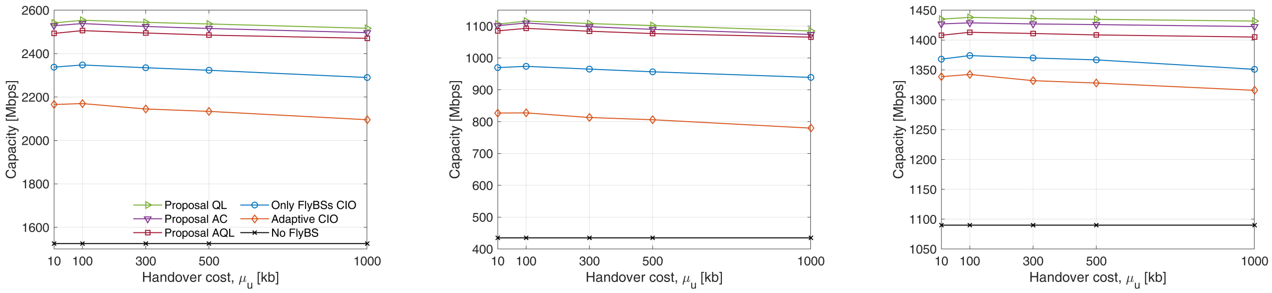
Based on the results in Figure 3, further evaluations of the *Proposal AQL* are performed for the reduced state space with $M = 4$ predetermined thresholds dividing the load of the BSs into five levels. This value of M leads to a significant reduction in the Q-table size (from 100^{K+1} states to 5^{K+1} states) while an impact on the capacity, HFR, and HPR is still marginal.

2) Impact of handover cost on capacity: Furthermore, let us demonstrate an impact of the handover cost on the capacity for all variants of the proposals as well as for competitive algorithms in Figure 4. Individual subplots show the capacity of all UEs (subplot a), the UEs served by the FlyBSs (subplot b), and the UEs served by the GBSs (subplot c). For a low handover cost, no negative impact on the sum capacity is observed, since the low handover overhead is compensated by an increase in the capacity of the handovering UE(s). However, for a higher handover cost, a decrease in the sum capacity is observed with a 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 [50], the handover overhead typically ranges in order of dozens to hundreds kb per UE per handover. Hence, for further analysis, we select μ_u =100 kb. Note that Figure 4 confirms that the relative gains introduced by the proposed algorithms with respect to state-of-the-art work are almost independent of the selected handover cost.

3) Capacity of UEs: Now, we demonstrate an impact of c_{req} on the capacity of the UEs for our proposals and the state-of-the-art algorithms in Figure 5. Individual subplots show the capacity of all UEs (subplot a), the UEs served by the FlyBSs (subplot b), and the UEs served by the GBSs (subplot c). The proposed algorithms *Proposal AQL*, *Proposal AC*, and *Proposal QL* outperform the state-of-the-art *Adaptive CIO* by up to 15.6%, 17% and 17.5%, respectively; and also our prior work *Only FlyBSs CIO* by up to 6.7%, 8.1% and 8.5%, respectively, in terms of the capacity of all UEs (see Figure 5). This increase in the UEs' capacity is because the proposed algorithm prevents the BSs' overloading and fairly distributes the FlyBSs among the GBSs and the UEs among the BSs by setting different CIO for all BSs. The *Proposal*

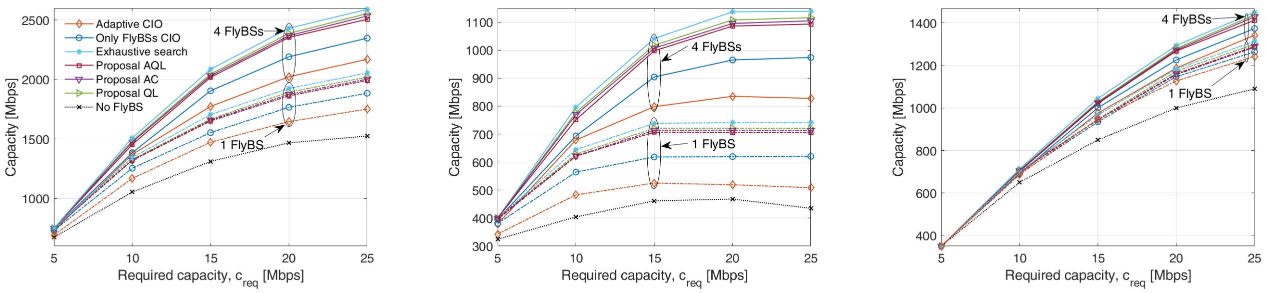
© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342



(a) (b) (c)

Fig. 4. Impact of handover cost on capacity of: (a) all UEs; (b) only UEs served by four FlyBSs ; (c) only UEs served by GBSs.



(a) (b) (c)

Fig. 5. Impact of c_{req} on capacity of: (a) all UEs; (b) only UEs served by FlyBSs; (c) only UEs served by GBSs; for handover cost = 100 kb.

QL provides a slightly higher capacity in comparison to the *Proposal AQL* and the *Proposal AC*. The capacity achieved by the *Proposal AC* is only 1% below the *Proposal QL*, since the actor-critic approach can mimic well the behavior of the complete Q-table considered in the *Proposal QL*. In the *Proposal AQL*, the quantization of the load of BSs into 5 ($M + 1$) levels introduces a quantization noise, which slightly degrades the capacity (by up to 2% compared to *Proposal QL*). Still, all variants of the proposal reach capacity close to the upper bound determined via the *Exhaustive search* with only a marginal difference lower than 3.4%, 2.5%, and 1.9% for the *Proposal AQL*, *Proposal AC*, and *Proposal QL*, respectively.

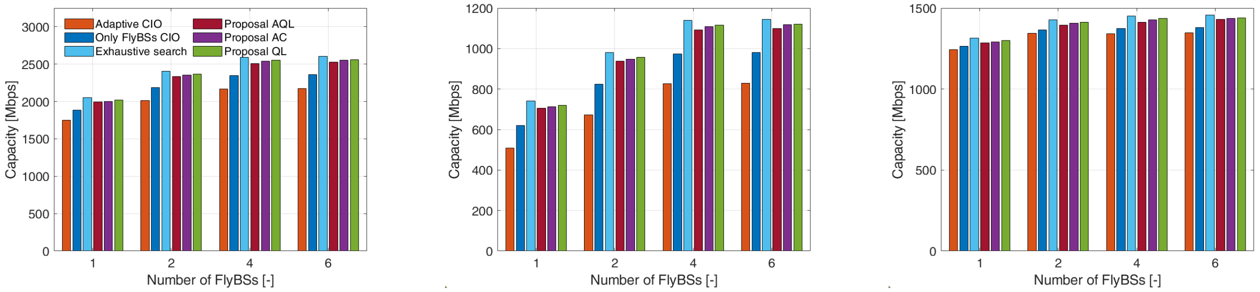
As our main objective is to optimize the handover decisions for the FlyBSs, we demonstrate the capacity of UEs served by FlyBSs in Figure 5b. For all algorithms, the capacity of UEs served by FlyBSs raises with c_{req} up to roughly c_{req} = 20 Mbps. Then, for c_{req} higher than 20 Mbps, the capacity becomes almost constant or even starts slightly decreasing. This saturation and/or decrease in the capacity of the UEs is a result of the limited bandwidth of the GBS. Hence, while all UEs (served by any BS) require a higher capacity, the GBSs still have only the same bandwidth that can be allocated. However, the proposed algorithm outperforms both the *Adaptive CIO* and our prior work *Only FlyBSs CIO* for all c_{req} values. For four FlyBSs, the proposed algorithms *Proposal AQL*, *Proposal AC*, and *Proposal QL* increase the capacity by up to 32.5%, 34%, and 34.8%, respectively, compared to the *Adaptive CIO* and by up to 12.2%, 14%, and 14.6%, respectively, compared to the *Only FlyBSs CIO* algorithms. The proposed algorithms *Proposal AQL*, *Proposal AC*, and *Proposal QL* achieve the capacity close to the upper bound (determined via *Exhaustive search*) with up to 4.6%, 3.5%, and 2.5% degradation, respectively.

To demonstrate that the proposed algorithms do not have a negative impact on the UEs served by the GBSs, in Figure 5c, we show the capacity of the UEs served only by the GBSs. The UEs' capacity for all compared algorithms is similar and the proposal even slightly increases the capacity of the UEs attached to the GBSs by up to 6% and 4% compared to the *Adaptive CIO* and the *Only FlyBSs CIO* algorithms, respectively. The capacity loss of our proposals compared to the *Exhaustive search* is always lower than 2%.

Next, lets investigate an impact of the number of FlyBSs on the capacity of UEs for our proposal and the competitive state-of-the-art algorithms in Figure 6. Individual subplots, again, show the capacity of all UEs (subplot a), the UEs served by the FlyBSs (subplot b), and the UEs served by the GBSs (subplot c). The proposed algorithm outperforms the *Adaptive CIO* and *Only FlyBSs CIO* algorithms in the capacity disregarding the numbers of the FlyBSs. The relative gain in the capacity for all UEs (Figure 6a) achieved by the proposed algorithm with respect to the state-of-the-art algorithms even slightly increases with the number of the FlyBSs, since the proposed algorithm prevents the FlyBSs and the UEs from connecting to the same GBS simultaneously to avoid the overloading of the GBSs. The achieved capacity starts saturating for a higher number of the FlyBSs due to the limited amount of bandwidth and the interference among the FlyBSs and the GBSs, since the

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342



(a)

(b)

(c)

Fig. 6. Impact of number of FlyBSs on capacity of: (a) all UEs; (b) only UEs served by FlyBSs; (c) only UEs served by GBSs; for $c_{req} = 25$ Mbps and handover cost = 100 kb.

additional FlyBSs increase interference level in the system. For six FlyBSs, the proposed algorithms *Proposal AQL*, *Proposal AC*, and *Proposal QL* increase the capacity of all UEs by up to 16.3%, 17.3%, and 17.7%, respectively, compared to the *Adaptive CIO* and by up to 7.2%, 8.1%, and 8.6%, respectively, compared to the *Only FlyBSs CIO*. The capacity achieved by the proposals is close to the upper bound reached by the *Exhaustive search* with the difference always below 3.5% for all UEs regardless the number of FlyBSs.

Figure 6b show the proposed algorithms *Proposal AQL*, *Proposal AC*, and *Proposal QL* increase the capacity of the UEs served by the FlyBSs by up to 32.5%, 34%, and 35%, respectively, compared to the *Adaptive CIO* and by up to 12.3%, 14%, and 14.6%, respectively, compared to the *Only FlyBSs CIO*. The increase in the capacity of the UEs served by the FlyBSs achieved by the proposed algorithm is a result of a proper CIO setting for all BSs and a fair distribution of the FlyBSs among the GBSs.

Last, Figure 6c confirms that the gains in the capacity of the UEs served by the FlyBSs (demonstrated in Figure 6b) is not at the cost of the capacity of the UEs served by the GBSs. Our proposals even slightly (by 2-6%) increase the capacity of the UEs served by the GBSs compared to the state-of-the-art *Adaptive CIO* and *Only FlyBSs CIO* algorithms regardless of the number of FlyBSs. The proposed algorithms achieve the capacity close to the upper bound with only an insignificant degradation in a range of 0.8-2.2%.

The capacity gain achieved by our proposal slightly increases with the number of FlyBSs in the system. However, this statement is not valid for the *Exhaustive search*. The capacity gain introduced by the *Exhaustive search* compared to the proposals almost does not change and even slightly decreases with the raising number of FlyBSs. The slight improvement in performance for our proposed solutions is because the proposals prevent the BSs' overloading and also suppress an occurrence of the redundant handovers. These two factors become critical in larger networks with a high number of BSs.

4) Learning process: Figure 7 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 in the capacity of all UEs achieved by the proposal with respect to the *Adaptive CIO* (subplot a) and *Only FlyBSs CIO* (subplot b) algorithms for four FlyBSs. At the beginning of the learning process, the gain of the proposed algorithms compared to the *Adaptive CIO* and *Only FlyBSs CIO* is rather small or even slightly negative (several percent) in some steps. 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, the gain becomes always non-negative. This initial phase, in Figure 7 represented by red vertical lines, lasts only about 30 – 40 handovers.

The figure also illustrates fitting function for the gain with respect to the *Adaptive CIO* (Figure 7a) and *Only FlyBSs CIO* (Figure 7b) algorithms. The fitting functions in Figure 7 show that the *Proposal AC* converges faster than the *Proposal QL* and the *Proposal AQL* and reaches convergence after approximately 100 handovers, while the *Proposal QL* reaches convergence after 160 handovers and the *Proposal AQL* converges after approximately 130 handovers. Note that the convergence is depicted by blue vertical line in the figures. The fast convergence of the actor-critic based approach is because the actor-critic does not utilize the Q-table and circumvents the long learning problems. The slower convergence of the *Proposal QL* is due to the big Q-table, which requires more time to fill and explore the whole table. For the similar reason, the *Proposal AQL* with approximate Q-table converges faster compared to the *Proposal QL* and converges slower compared to the *Proposal AC*. Nevertheless, all proposals eventually converge to similar capacity gain with a marginal difference below 2%.

5) Handover failure and ping-pong ratios: Figure 8 depicts the handover failure ratio HFR for the handovers of the FlyBSs (subplot a) and for the handovers of the UEs (subplot b). The proposed algorithm always reaches the lowest HFR for the FlyBSs as well as for the UEs. The *Adaptive CIO*, *Only FlyBSs CIO*, and *Exhaustive search* algorithms lead to HFR equal to 4.3%, 2.7%, and 3.4%, respectively, for the FlyBSs (Figure 8a). In contrast, our proposed algorithms achieve HFR always below 0.5% for the FlyBSs, i.e., all proposals reduce HFR more than five times with respect to the state-of-the-art works. The difference in the achieved HFR for the FlyBSs and the UEs by *Proposal QL*, *Proposal AQL*, and *Proposal*

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

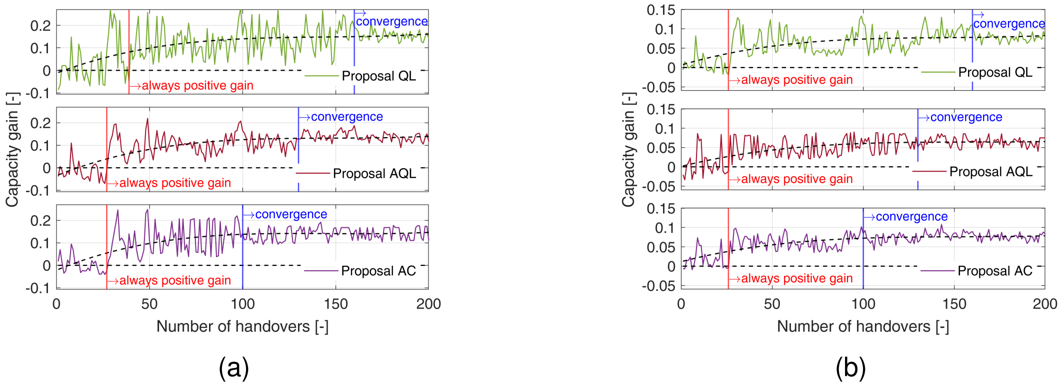


Fig. 7. Learning progress represented by gain in capacity over number of handovers performed by four FlyBSs for $c_{req} = 25$ Mbps with respect to: (a) *Adaptive CIO*; (b) *Only FlyBSs CIO*. Red vertical line corresponds to the point, after which the gain with respect to related works is always positive; blue vertical line depicts convergence of the proposed algorithms.

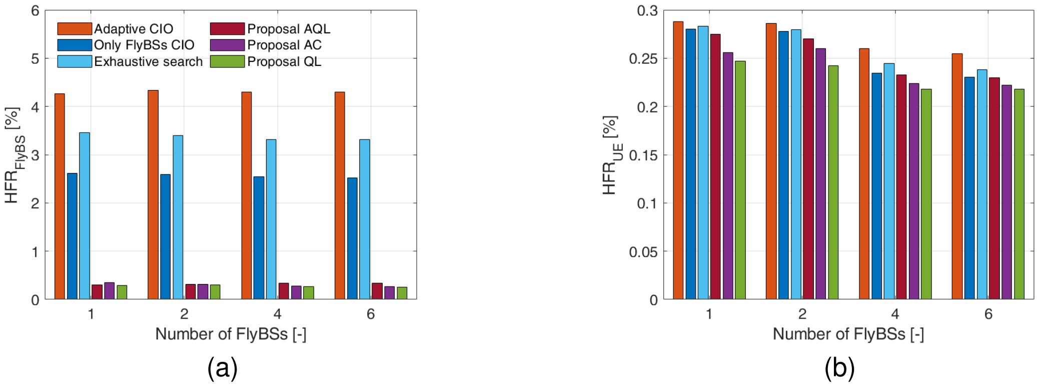


Fig. 8. Handover failure ratio for handovers of FlyBSs (a) and all UEs (b).

AC is below 0.03% and is negligible. The proposed algorithms eliminate the handover failures of the FlyBSs caused by GBSs' overloading by adjusting the CIO of the BSs.

The HFR of the UEs (Figure 8b) is always below 0.3% for all compared algorithms. Still, all three variants of the proposed algorithm reach a bit lower HFR than the related state-of-the-art works. We observe that the HFR for the UEs slightly decreases with an increasing number of the FlyBSs in the network. The reason for this slight decrease in the HFR with more FlyBSs is the higher number of BSs to which the UEs can connect in case of a low signal quality from the serving BS and by overall improvement in SINR in the system leading to a lower probability of the handover failure due to low SINR.

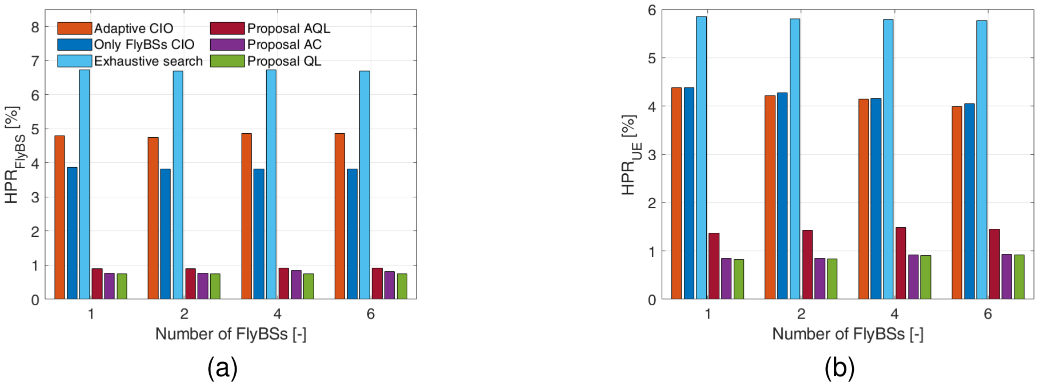
In Figure 9, we show the HPR of all algorithms for the handovers performed by the FlyBSs (subplot a) and the handovers performed by the UEs (subplot b). The HPR for the FlyBSs achieved by the *Exhaustive search*, the state-of-the-art *Adaptive CIO*, and our prior work *Only FlyBSs CIO* is approximately 6.7%, 5%, and 3.9%, respectively. In contrast, all three proposed algorithms significantly decrease the HPR for the FlyBSs below 1% (Figure 9a), i.e., roughly seven-, five-, and four- times compared to the *Exhaustive search*, *Adaptive CIO*, and *Only FlyBSs CIO* algorithms, respectively. The HPR for the UEs (Figure 9b) reached by the *Adaptive CIO* and the *Only FlyBSs CIO* is above 4% while our *Proposal QL*, *Proposal AQL*, and *Proposal AC* reduce the HPR of the UEs below 1%, 1.5%, and 1%, respectively, i.e., roughly three to four times. The proposed algorithm reduces the number of performed handovers by preventing the BS's overloading and taking the handover cost into account.

V. Conclusions

In this paper, we have proposed a novel algorithm simultaneously managing the handover of both FlyBSs and UEs to maximize the capacity of the UEs served by the FlyBSs while taking the handover cost into account. The proposed algorithm exploits three variants of reinforcement learning to adjust CIO of all BSs (including GBSs and FlyBSs) for each FlyBS and each UE according to the load of BSs and the load generated by FlyBSs and UEs. The three variants provide a trade-off between performance and requirements related to a practical implementation. The first proposal is based on a common tabular Q-learning and yields the highest capacity close to the theoretical upper bound and reaches also the lowest HPR and HFR out of all proposals. However, this common approach also implies notable requirements on computation and storage for the Q-table and the Q-table even

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342



(a)

(b)

Fig. 9. Handover ping-pong ratio for: (a) FlyBSs; (b) all UEs.

significantly enlarges with the raising number of BSs in the system resulting into a longer learning period. In order to solve the problem of the large Q-table, we further propose a solution based on the approximate Q-table, which limits the practical implementation issues at a cost of a small decrease in the capacity (few percent) and a small increase in both HPR and HFR. Still, even the approximate Q-table can be limiting in large scale mobile networks. Hence, we propose the actor-critic deep reinforcement learning approach. The actor-critic-based solution is seen as the most suitable for the practical implementation, since it reaches performance close to the common tabular Q-learning in all investigated metrics and, at the same time, the actor-critic reduces the learning period and completely avoids the practical implementation problems with the Q-table size.

The results show that all proposed reinforcement learning based approaches outperform the state-of-the-art solutions in the UEs' capacity by dozens of percent and, at the same time, the proposed algorithms also reduce both HFR and HPR for the FlyBSs as well as for the UEs multiple times and reduce both metrics to a negligible level.

References

[1] X. Li, H. Yao, J. Wang, X. Xu, C. Jiang and L. Hanzo, ”A Near-Optimal UAV-Aided Radio Coverage Strategy for Dense Urban Areas,” in IEEE Transactions on Vehicular Technology, vol. 68, no. 9, pp. 9098-9109, Sept. 2019.

[2] A. S. Khan, G. Chen, Y. Rahulamathavan, G. Zheng, B. Assadhan and S. Lambotharan, ”Trusted UAV Network Coverage Using Blockchain, Machine Learning, and Auction Mechanisms,” in IEEE Access, vol. 8, pp. 118219-118234, 2020.

[3] M. Mozaffari, W. Saad, M. Bennis, Y. H. Nam and M. Debbah, ”A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems,” in IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2334-2360, thirdquarter 2019.

[4] 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.

[5] 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.

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

[7] 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.

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

[9] Y. Li, W. Wang, M. Liu, N. Zhao, X. Jiang, Y. Chen, X. Wang, ”Joint Trajectory and Power Optimization for Jamming-Aided NOMA-UAV Secure Networks”, *IEEE Systems Journal*, 2022.

[10] W. Belaoura, K. Ghanem, M. Z. Shakir and M. O. Hasna, ”Performance and User Association Optimization for UAV Relay-Assisted mm-Wave Massive MIMO Systems,” *IEEE Access*, vol. 10, 2022.

[11] N. Zhao et al., ”Joint Trajectory and Precoding Optimization for UAV-Assisted NOMA Networks,” *IEEE Transactions on Communications*, vol. 67, no. 5, May 2019.

[12] X. Pang, M. Sheng, N. Zhao, J. Tang, D. Niyato and K. -K. Wong, ”When UAV Meets IRS: Expanding Air-Ground Networks via Passive Reflection,” *IEEE Wireless Communications*, vol. 28, no. 5, October 2021.

[13] 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

[14] 3GPP, “E-UTRA radio resource control (RRC) protocol specification (Release 8),” 3GPP, Tech. Rep. 36.331 V16.3.0, Jan. 2021.

[15] G. Alsuhli, K. Banawan, K. Seddik and A. Elezabi, ”Optimized Power and Cell Individual Offset for Cellular Load Balancing via Reinforcement Learning,” *IEEE Wireless Communications and Networking Conference (WCNC)*, 2021.

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

[17] 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.

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

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

[20] J. Shodamola, U. Masood, M. Manalastas and A. Imran, ”A Machine Learning based Framework for KPI Maximization in Emerging Networks using Mobility Parameters,” *IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, 2020.

[21] M. -S. Pan, T. -M. Lin and W. -T. Chen, ”An Enhanced Handover Scheme for Mobile Relays in LTE-A High-Speed Rail Networks,” in *IEEE Transactions on Vehicular Technology*, vol. 64, no. 2, pp. 743-756, Feb. 2015.

[22] M. Tayyab, G. P. Koudouridis, X. Gelabert and R. Jäntti, ”Uplink Reference Signals for Power-Efficient Handover in Cellular Networks With Mobile Relays,” in *IEEE Access*, vol. 9, pp. 24446-24461, 2021.

[23] X. Qian and H. Wu, ”Mobile Relay Assisted Handover for LTE System in High-Speed Railway,” *International Conference on Control Engineering and Communication Technology*, 2012.

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

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

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

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2022.3216342

[27] J. Bai, S.-P. Yeh, F. Xue, and S. Talwar, "Route-aware handover enhancement for drones in cellular networks," *IEEE GLOBECOM*, 2019.

[28] Y. Jang, S. M. Raza, H. Choo and M. Kim, "UAVs Handover Decision using Deep Reinforcement Learning," *International Conference on Ubiquitous Information Management and Communication (IMCOM)*, 2022.

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

[30] A. Madelkhanova, Z. Becvar, T. Spyropoulos, "Q-Learning-based Adjustment of Cell Individual Offset for Handover of Flying Base Stations," *IEEE VTC - Spring*, 2022.

[31] 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.

[32] 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.

[33] M. Tayyab, G. P. Koudouridis, X. Gelabert and R. Jäntti, "Signaling Overhead and Power Consumption during Handover in LTE," *IEEE Wireless Communications and Networking Conference (WCNC)*, 2019.

[34] R. Arshad, H. Elsawy, S. Sorour, T. Y. Al-Naffouri and M. Alouini, "Handover Management in 5G and Beyond: A Topology Aware Skipping Approach," in *IEEE Access*, vol. 4, pp. 9073-9081, 2016.

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

[36] V. R. Konda and J. N. Tsitsiklis, "Actor-critic algorithms," in Proc. Adv. Neural Inf. Process. Syst., 2000, pp. 1008–1014.

[37] J. Peters and S. Schaal, "Natural actor–critic," Neurocomputing, vol. 71, nos. 7–9, pp. 1180–1190, Mar. 2008.

[38] C. J. C. H. Watkins and P. Dayan, "Q-learning," Mach. Learn., vol. 8, nos. 3–4, pp. 279–292, May 1992.

[39] V. Mnih et al., "Asynchronous methods for deep reinforcement learning," in Proc. Int. Conf. Mach. Learn., New York, NY, USA, Feb. 2016, pp. 1928–1937.

[40] I. Grondman, L. Busoniu, G. A. D. Lopes and R. Babuska, "A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 6, pp. 1291-1307, Nov. 2012

[41] C. Bettstetter, G. Resta and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," in IEEE Transactions on Mobile Computing, vol. 2, no. 3, pp. 257-269, July-Sept. 2003

[42] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks", *MSWiM*, 1999.

[43] Z. Becvar, P. Mach, J. Plachy and M. F. P. de Tudela, "Positioning of Flying Base Stations to Optimize Throughput and Energy Consumption of Mobile Devices," *VTC-Spring*, 2019

[44] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Mobility enhancements in heterogeneous networks (Release 11)," 3GPP, Tech. Rep. 36.839 V11.1.0, Jan. 2013.

[45] A. Al-Hourani, S. Kandeepan and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," IEEE Wireless Communications Letters, vol. 3, no. 6, pp. 569-572, Dec. 2014.

[46] R. I. Bor-Yaliniz, A. El-Keyi and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," *IEEE International Conference on Communications (ICC)*, Kuala Lumpur, 2016.

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

[48] Y. Chen, K. Niu and Z. Wang, "Adaptive Handover Algorithm for LTE-R System in High-Speed Railway Scenario," in IEEE Access, vol. 9, pp. 59540-59547, 2021

[49] Technical Specification Group Radio Access Network; Mobility Enhancements in Heterogeneous Networks (Release 11), document TR 36.839V11.1.0, 3rd Generation Partnership Project, Dec. 2012.

[50] 3GPP, "Handover Procedures," 3GPP, TS 23.009 V16.0.0, Jul. 2020.

Aida Madelkhanova received the BSc degree in Communication Technologies from the Czech Technical University in Prague, Czech Republic in 2017. She received MSc degree from CTU in Prague, Czech Republic and from EURECOM, France in Telecommunications engineering within Double-Degree Program in 2020. Currently, she works towards the Ph.D. degree at the Department of Telecommunication Engineering at the Czech Technical University in Prague with a topic Mobility management in cellular networks with flexible architecture of radio access network.

Zdenek Becvar (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in telecommunication engineering from the Czech Technical University in Prague, Czech Republic, in 2005 and 2010, respectively. He is currently an Associate Professor with the Department of Telecommunication Engineering, Czech Technical University in Prague. From 2006 to 2007, he joined Sitronics R&D Center, Prague, focusing on speech quality in VoIP. Furthermore, he was involved in research activities of Vodafone R&D Center, Czech Technical University in Prague, in 2009. He was on internships at Budapest Politechnic, Hungary, in 2007; CEA-Leti, France, in 2013; and EURECOM, France, in 2016 and 2019. From 2013 to 2017, he was a representative of the Czech Technical University in Prague in ETSI and 3GPP standardization organizations. In 2015, he founded 5Gmobile Research Lab at CTU in Prague, focusing on research towards 5G and beyond mobile networks. He has published four book chapters and more than 70 conference or journal papers.

Thrasyvoulos Spyropoulos (Member, IEEE) received the Diploma degree in electrical and computer engineering from the University of Athens, and the Ph.D. degree from the University of Southern California. He was a Post-Doctoral Researcher with INRIA and then a Senior Researcher with ETH Zürich. He is currently a Professor at EURECOM, Sophia Antipolis. He was a recipient of the Best Paper Award in IEEE SECON 2008 and IEEE WoWMoM 2012, and runner-up for ACM Mobihoc 2011 and IEEE WoWMoM 2015.

© 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: CZECH TECHNICAL UNIVERSITY. Downloaded on December 05,2022 at 12:28:14 UTC from IEEE Xplore. Restrictions apply.