Multi-UAV Dynamic Wireless Networking With Deep Reinforcement Learning

Qiang Wang, Wenqi Zhang, Yuanwei Liu, and Ying Liu

***Abstract*— This letter investigates a novel unmanned aerial vehicle (UAV)-enabled wireless communication system, where multiple UAVs transmit information to multiple ground terminals (GTs). We study how the UAVs can optimally employ their mobility to maximize the real-time downlink capacity while covering all GTs. The system capacity is characterized, by optimizing the UAV locations subject to the coverage constraint. We formula the UAV movement problem as a Constrained Markov Decision Process (CMDP) problem and employ Q-learning to solve the UAV movement problem. Since the state of the UAV movement problem has large dimensions, we propose Dueling Deep Q-network (DDQN) algorithm which introduces neural networks and dueling structure into Q-learning. Simulation results demonstrate the proposed movement algorithm is able to track the movement of GTs and obtains real-time optimal capacity, subject to coverage constraint.**

***Index Terms*— Capacity, deep reinforcement learning, movement, unmanned aerial vehicles.**

I. Introduction

UNMANNED aerial vehicle (UAV) is also known as drone, has been sought after by many researchers because of its potential in future applications [1]. Due to the controllable mobility of drones, drones can be used as mobile base stations to improve the performance of terrestrial wireless networks, such as, temporary aerial base stations for terrestrial emergency rescue [2]. In particular, compared with traditional fixed communication structures, drones can timely adjust their locations according to the movement of ground users (GTs). To fully exploit the advantage of drones, movement issue is significant for UAV-aided wireless networks.

The movement issue of drones is worth studying since it greatly affects the performance of dynamic wireless networks. Nevertheless, the exiting works mainly consider the UAV movement problem with static ground users. For example, in [3], the authors designed the trajectory of a single UAV to research the energy-efficient maximization problem for the wireless networks with one ground user. Finally, the trajectory of UAV is obtained with the general trajectory constraints (i.e. the initial location of UAV, the final location of UAV and the speed of UAV). [4] proposed an effective solution for moving a single UAV to maximize the minimum average throughput of all GTs. However, multiple-UAV case not be considered in this letter. The problem of multiple-UAV dynamic movement based on the moving GTs is still an open question, the content of [3] and [4] offered inspiration into what we investigated in this letter.

The main contribution of this letter is that, optimally exploit the UAVs' mobility with the proposed Dueling Deep Q-network (DDQN) algorithm to improve the performance of system. Given a target geographical area, the movement issue of UAVs can be modeled as a Constrained Markov Decision Process (CMDP) problem. CMDP is a typical framework for reinforcement learning tasks with constraints, where agents learn an action policy that maximizes the long-term reward while satisfying the constraints on the long-term cost [5]. For our proposed movement algorithm, we introduce neural networks into Q-learning, which is referred as deep reinforcement learning [6] (also represented as Deep Q-network (DQN)). As a further advance, we adopt dueling structure for action selection of each DQN state in order to improve the performance of DQN. Moreover, in our proposed algorithm, each agent (UAV) learns from the experience samples which randomly selected from "replay memory" and automatically updates the weights of the neural networks within its movement machine to obtain an efficient movement strategy. In a nutshell, each drone tries and studies the movement to adapt the movement of GTs.

II. System Model

Consider a square geographical area of side length D divided into $F \times F$ small grids and N drones used to serve all moving ground terminals in the square geographical area. Each drone can move in a straight line between current grid and its adjacent grids (i.e., its front, behind, right and left adjacent grids). And all drones timely move to adapt the change of network topology, which maximizes the total capacity while covering all GTs. Moreover, we assume that all drones have the same constant altitude h, which is the minimum altitude according to safety consideration.

A. Air-to-Ground Communications Model

In this letter, we consider the Air-to-Ground (AtG) channel model as in [7]. A line-of-sight (LoS) dominating environment is considered. Since, in sub-urban or rural environment, the line-of-sight links dominate any other links [8]. Furthermore, the multi-path (small scale fading) links are also considered in this AtG channel model. Thus, the pathloss is

$$PL = (\frac{4\pi f_c}{c})^{-2}(d_{mn})^{-\alpha}|\varphi_{mn}|^2, \quad (1)$$

where f_c denotes the carrier frequency, c denotes the speed of light, α denotes pathloss exponent ($\alpha \geq 2$), d_{nm} denotes

Manuscript received July 26, 2019; revised August 26, 2019; accepted August 30, 2019. Date of publication September 11, 2019; date of current version December 10, 2019. This work is supported by Beijing Natural Science Foundation No.L182037 and National Natural Science Foundation of China No.61871045, in part supported by Beijing Natural Science Foundation under Grant L172033, the National Natural Science Foundation of China (6197106661325006) and the 111 Project of China B16006. The associate editor coordinating the review of this letter and approving it for publication was H. Tabassum. *(Corresponding author: Qiang Wang.)*

Q. Wang, W. Zhang, and Y. Liu are with the Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: wangq@bupt.edu.cn; zwqximo@bupt.edu.cn; yliu@bupt.edu.cn).

Y. Liu is with the Queen Mary University of London, London E1 4NS, U.K. (e-mail: yuanwei.liu@qmul.ac.uk).

Digital Object Identifier 10.1109/LCOMM.2019.2940191

the spatial distance between drone n and ground terminal m, is given by $\sqrt{h^2 + l_{nm}^2}$, where l_{nm} denotes the horizontal distance between UAV n and ground terminal m. In addition, φ_{mn} follows Rician fading between between drone n and ground terminal m, and

$$\varphi_{mn} = \sqrt{\frac{\eta}{1+\eta}}\varphi_L + \sqrt{\frac{1}{1+\eta}}\varphi_M, \tag{2}$$

where η is the ratio between the energy of LoS component and the energy of multi-path component, φ_L is a normalized constant of LoS component and φ_M is the circularly symmetric complex Guassian (CSCG) random variable with zero mean and unit variance.

B. Communication Channel Model

We consider a multi-drone downlink wireless network which consists of N drones and M GTs, denoted by $N = \{1, 2, \ldots, n, \ldots, N\}$ and $\boldsymbol{M} = \{1, 2, \ldots, m, \ldots, M\}$, respectively. The drones and GTs are equipped with a single antenna. The system employs the universal frequency reuse deployment in which each drone shares the whole resource. And the resource is divided into K subchannels, denoted by $\boldsymbol{K} = \{1, 2, \ldots, k, \ldots, K\}$. Each GT is connected to one drone and occupies one subchannel of the drone. In this letter, we allocate subchannels to GTs as the method described in [9] and allocate power to the drones as the method described in [10]. Here, we define U_{nk}^m as the indicator of subchannel, where $U_{nk}^m = 1$ if the k-th subchannel of drone n is occupied by GT m; $U_{nk}^m = 0$, otherwise. Thus, the signal-to-interference-plus-noise ratio (SINR) for wireless communication between drone n and GT m over subchannel k is calculated as

$$r_{nk}^m = \frac{P_{nm} \cdot PL_{nm} \cdot U_{nk}^m}{I_{nk}^m + \sigma^2}. \tag{3}$$

where P_{nm} is transmit power from drone n to GT m, PL_{nm} is path loss between drone n and GT m, which is described in Air-to-Ground Communications Model part. I_{nk}^m is interference to GT m with $I_{nk}^m = \sum_{j \in N, j \neq n} \sum_{i \in M, i \neq m} P_{ni} \cdot U_{jk}^i \cdot PL_{ji}$. Therefore, the capacity between drone n and GT m over subchannel k can be expressed as

$$C_{nk}^m = \frac{W}{K} \log_2(1 + r_{nk}^m), \tag{4}$$

where W is resource. Apparently, the total downlink capacity of the system is

$$C_{capacity} = \sum_{n \in N} \sum_{k \in K} \sum_{m \in M} C_{nk}^m. \tag{5}$$

For coverage, we define O_{nk}^m as the indicator of coverage, where $O_{nk}^m = 1$ if the SINR ratio for wireless communication between drone n and GT m over subchannel k is greater than threshold $\bar{\kappa}$; $O_{nk}^m = 0$, otherwise. Thus, the coverage of the system is calculated as

$$C_{coverage} = \sum_{n \in N} \sum_{k \in K} \sum_{m \in M} O_{nk}^m. \tag{6}$$

III. Efficient Movement of Multiple UAVs

In our proposed movement algorithm, all drones have to explore new locations in a real-time manner to adapt the movement of GTs. For each exploration, each drone choose an adjacent grid which it will move to. After making the choice, the drone is able to calculate the new location's capacity and number of covered GTs. As a further advance, the new location's capacity and number of covered GTs are used to judge the impact of this choice on the entire network's performance.

A. Problem Formulation

We obtain system capacity and coverage from Section II. Hereafter, the objective is to maximize real-time capacity $C_{capacity}(t)$ with constraint $C_{coverage}(t) = M$. The drone moving problem is to determine the locations of drones aiming to maximize downlink capacity, subject to the coverage constraint. Thus, the drone moving problem is expressed as

$$\max_{p_1, p_2, \ldots, p_N} \sum_{n \in N} \sum_{k \in K} \sum_{m \in M} C_{nk}^m \tag{8a}$$

$$s.t.\ C_1 : \sum_{n \in N} \sum_{k \in K} U_{nk}^m = 1, \tag{8b}$$

$$C_2 : \sum_{k \in K} \sum_{m \in M} U_{nk}^m \leq K, \tag{8c}$$

$$C_3 : r_{nk}^m \geq \bar{\kappa}, \quad \forall n \in \boldsymbol{N},\ \forall m \in \boldsymbol{M},\ \forall k \in \boldsymbol{K}, \tag{8d}$$

$$C_4 : C_{coverage}(t) = M, \tag{8e}$$

where p_n denotes the location of drone n, $n = 1, 2, 3, \ldots, N$. (8b) states a GT can only occupy one subchannel and (8c) states one drone is able to serve at most K GTs. (8d) ensures the SINR ratio for wireless communication between drone n and GT m over subchannel k should be greater than threshold. (8e) is coverage constraint. Given subchannel and power allocation, the sum capacity maximization problem is simplified into UAV movement optimal problem. Since the movement of GTs affect the sum capacity and coverage of the system, the drones have to change their locations in a real-time manner to adapt the movement of GTs, the problem of maximizing the sum capacity in (8a) inherently incorporates the real-time movement of drones.

Since the next location of drone is only rely on the current location of drone and the UAV movement decisions, the multi-drone movement problem is a Markov Decision Process (MDP) problem. Furthermore, the policy should satisfy the coverage constraint, so the problem in (8) is a CMDP problem and can be transformed into the following form

$$\min_{\lambda \geq 0} \max_{\pi} C_{capacity}(t) - \lambda(M - C_{coverage}(t)), \tag{9}$$

where λ is the Lagrangian multiplier and π is policy. Since the multi-drone movement problem is the generalization of large dimension MDP, we employ DQN to solve the problem. Furthermore, in order to improve the performance of DQN, we adopt dueling structure for action selection of each DQN state. We call the improved algorithm as DDQN.

B. DDQN for Movement of Multiple UAVs

In DDQN model, the drones act as agents, and DDQN is composed of state, action, reward, cost and state-action value (Q-value). DDQN aims to obtain the optimal policy (a series of actions) that maximizes the long-term accumulated reward

(with constraint cost) starting from the initial state. At each time, each drone observes a state from state space. And, each drone takes an action from action space, according to policy. The principle of policy is to choose an action that maximizes the Q-value of each time. After taking actions, each drone get a reward and a cost, which determined by instantaneous sum capacity and coverage of GTs respectively.

Here, we denote the $state$ for UAV n at time t by $s_n(t) = (\bar{s}_n(t), g_n(t))$, where $g_n(t)$ is the location information of all UAVs except UAV n. And $\bar{s}_n(t) \in \{(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots, (x_{F\times F}, y_{F\times F})\}$, where (x_i, y_i) is the location of the center of grid i and the state represents the location which the drone can move to. The $action$ by UAV n adopting the movement algorithm at time t is $a_n(t) \in \{$FRONT, BEHIND, RIGHT, LEFT, HOVER$\}$ where FRONT, BEHIND, RIGHT, LEFT mean that the drone moves to the front, behind, right and left adjacent grid, respectively. Furthermore, HOVER means the drone hovers in the current grid. After taking action $a_n(t)$, the transition from state $s_n(t)$ to state $s_n(t + t_\delta)$ generates reward $r_n(t) = (C_{capacity}(t + t_\delta) - C_{capacity}(t))/N$ and cost $c_n(t) = (C_{coverage}(t + t_\delta))/N$, where t_δ is the time required for a drone to fly from the current grid to an adjacent grid.

DQN is a value-based reinforcement learning algorithm. Moreover, DQN uses neural network to store state and action information $Q^\pi(s, a)$. The objective of DQN is to seek out the optimal policy $\pi^\star$, and obtain optimal state-action value $Q^\star(s, a)$, which is expressed as $\pi^\star(s) = arg\max_a Q^\star(s, a)$. According to the drone moving problem in this letter, the global state-action value $Q(s, a)$ is a linear combination of all local state-action values $Q_n(s_n, a_n)$, $Q(s, a) = \sum_1^N Q_n(s_n, a_n)$. Thus, the global state-action value maximization is equal to the local state-action value maximization:

$$Q_n(s_n, a_n) = r_n + \gamma \max_{a'_n} Q_n(s'_n, a'_n) - \lambda(M/N - c_n), \quad (10)$$

where r_n is reward, c_n is cost, s_n is state, a_n is action, s'_n is next state, a'_n is next action. And γ is discount factor, which determines the balance between current state-action value and future state-action value. According to (10), we need a neural network, Q evaluation network, to estimate $Q_n(s_n, a_n)$. In order to drop the sample continuity, a separate neural network, "quasi-static target network", is used to train the Q target network. The target neural network will not change in every train step and will be updated after multiple train steps. For evaluation state-action value update, Bellman Equation is applied

$$Q^e_n(s_n, a_n) = (1-\Gamma)Q^e_n(s_n, a_n) + \Gamma(r_n + \gamma \max_{a'_n} Q^{tar}_n(s'_n, a'_n) - \lambda(M/N - c_n)), \quad (11)$$

where Q^e_n and Q^{tar}_n are the output of Q evaluation and target network, respectively. Γ denotes learning rate, which determines the rate of learning. And loss $Q^{tar}_n(s_n, a_n) - Q^e_n(s_n, a_n)$ is used to update the weights of Q evaluation network. For target state-action value update, $Q^{tar}_n(s_n, a_n) = Q^e_n(s_n, a_n)$. And we use the weights of Q evaluation network to update the weights of Q target network. We obtain the optimal policy (action) and Q value when the weights of Q evaluation and target network converge. In other words, drones find the locations which maximize capacity while covering all GTs.

Algorithm 1 DDQN for Multi-Drone Moving Problem (Training Stage)

Input: Initial locations of drones.
for $t = t_\delta, 2t_\delta, \ldots, \upsilon t_\delta$ **do**
  for *each drone* **do**
    Select a greedy action $a_n(t) = \arg\max_{a_n(t)} Q_n(s_n, a_n)$ with probability $1 - \epsilon$, and selects a random action with probability ϵ;
    Execute action $a_n(t)$ to observe the transition $(s_n(t), a_n(t), r_n(t), c_n(t), s_n(t + t_\delta))$ and store it in replay memory G;
    Sample random minibatch of transitions from replay memory;
    Update Q evaluation network;
    Update the Lagrangian multiplier with $\lambda = \lambda + \Gamma \frac{1}{|G|} \sum_{i=1}^{|G|} c^i_n$;
    Update Q target network, periodically
Output: The drone locations of each time.

Algorithm 2 DDQN for Multi-Drone Moving Problem (Testing Stage)

Input: Initial locations of drones.
for $t = t_\delta, 2t_\delta, \ldots, \upsilon t_\delta$ **do**
  for *each drone* **do**
    According to greedy algorithm select an action;
    Store $(s_n(t), a_n(t), r_n(t), c_n(t), s_n(t + t_\delta))$;
    Fine turning Lagrangian multiplier, Q target and evaluation network
Output: The drone locations of each time.

In order to improve the stability of DQN, we adopt dueling structure for action selection of each DQN state. The improved algorithm DDQN changes (10) into the combination of state value and action advantage value

$$Q_n(s_n, a_n) = V_n(s_n) + (A_n(s_n, a_n) - \frac{1}{|A_n|} \sum_{a^i_n \in A_n} A_n(s_n, a^i_n)), \quad (12)$$

where $V_n(s_n)$ is the value of state s_n and $|A_n|$ is the number of action for UAV n. Compare to DQN, DDQN can ensure that the relative order of advantage functions for all actions is unchanged under the same state, and can narrow the range of $Q_n(s_n, a_n)$ values. Therefore, DDQN removes redundant degrees of freedom and improves stability.

Details of DDQN algorithm for multi-drone moving problem are provided in Algorithm 1. In the multi-agent DDQN model, the learning needs to solve N Q-values. The complexity of handling one Q-value is $O(|S|^2|A|)$, therefore complexity of multi-agent DDQN model is $O(N|S|^2|A|)$. Obviously, the complexity of this model is increased polynomially with the number of state, linearly with the number of action. Then, the convergence is decided by the number of UAV. If the number of UAV is small, a faster convergence can be attained.

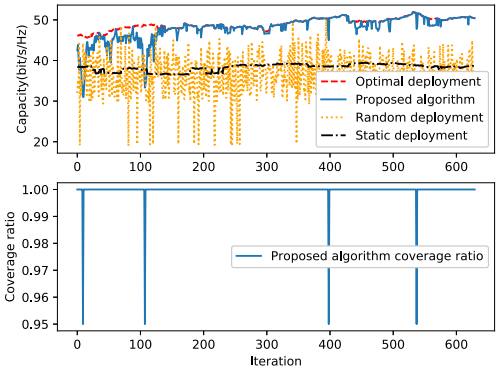

Fig. 1. For comparison, we will analyze the performance of optimal, static and random deployments. Optimal deployment refers to the real-time optimal grids (traversing method). Static deployment represents that drones are evenly distributed throughout the given area and suspend in the fixed locations. Random deployment means drones move randomly in the given area.

IV. Simulation Results

In our simulation, we consider a $1.5\,km \times 1.5\,km$ area. For AtG channel, we set $F = 6$, $f_c = 2\,GHz$, $\alpha = 2$, the altitude of drones $h = 100\,m$, $\varphi_L = 1$ and $\eta = 28\,dB$ [7]. For communication channel, we set resource 20 MHz, the number of subchannel $K = 10$, the maximum transmit power of drones is 23 dBm, the noise spectral density is $-170\,dBm/Hz$ and SINR threshold $\bar{\kappa} = 1$. We obtain the number of drones N according to $N = \lceil \frac{M}{K} \rceil$. In addition, we use a three-layer fully connected neural networks to estimate action-state value, and the hidden layer of the neural networks has 64 hidden neurons. ReLU function is used as activation function for neurons. Furthermore, the capacity of "replay memories" is 2000, and a batch of 128 empirical samples is randomly selected from "replay memories" to train the neural networks. In order to update weights, the Stochastic Gradient Descent is conducted by RMSProp algorithm. We employ $\epsilon = 0.1$ in ϵ-greedy algorithm. Furthermore, the rate of learning Γ is set to 0.01 and discount factor γ is set to 0.9.

In this simulation, ground terminals are dropped randomly with the density $\mu = 12\ GTs/km^2$. Moreover, we set the speed of the three drones and GTs are 25 m/s and 2 m/s, respectively. Our proposed algorithm converges after training (offline 30000 iterations) and we analyze the performance of our proposed algorithm with 600 test iterations. Fig. 1 illustrates the performance of our proposed movement algorithm versus optimal, static and random deployments. For our proposed algorithm, we give the drones a random initial location. At first, the drones spend some time adapting the test environment. Then, the drones track the GT movement and close to the optimal capacity since the speed of GTs is slow and drones can adjust their locations to adapt the GT movement. Furthermore, there are performance losses when the number of iterations are about 400 and 540 since the drones have ϵ probability to explore random locations. In order to adapt the movement of GTs, the drones should explore the new locations to indirectly obtain the movement information of GTs. And when the drones explore new locations, the new locations maybe have bad performance.

Here, we analyze how the velocity of GTs effects the performance of the proposed movement algorithm. In such a simulation, we set the velocity of all drones is 25 m/s and all GTs have the same absolute value of velocity. Fig. 2 illustrates the performance of the proposed movement algorithm versus different velocities of GTs. Clearly, for the same density of GTs, the high velocity scenarios have small sum capacity. This is because that the higher velocity causes the greater changes of the network topology. Furthermore, as the density of GTs increases, the capacity difference between the optimal sum capacity and the sum capacity of a specific velocity scenario increases. As expected, the drones have better performance in tracking slow moving GTs, since the change of network topology is small for low velocity scenarios.

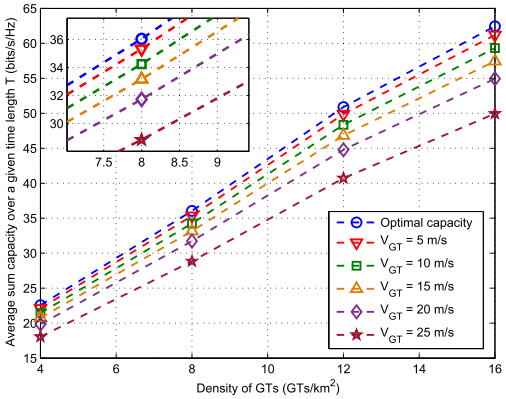

Fig. 2. The performance of the proposed movement algorithm versus different speeds of GTs.

V. Conclusion

In this letter, we investigated the efficient movement of multiple UAVs in order to maximize downlink capacity while covering all GTs. We showed that the problem can be modeled as a CMDP problem. Moreover, we developed a novel framework to determine the movement of the UAVs. Simulation results evaluated the performance of the proposed movement algorithm, where results showed that the proposed algorithm has good performance.

References

[1] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

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

[3] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.

[4] Q. Wu and R. Zhang, "Common throughput maximization in UAV-enabled OFDMA systems with delay consideration," *IEEE Trans. Wireless Commun.*, vol. 66, no. 12, pp. 6614–6627, Dec. 2018.

[5] Q. Liang, F. Que, and E. Modiano, "Accelerated primal-dual policy optimization for safe reinforcement learning," 2018, *arXiv:1802.06480*. [Online]. Available: https://arxiv.org/abs/1802.06480

[6] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[7] Z. Wang, L. Duan, and R. Zhang, "Adaptive deployment for UAV-aided communication networks," *IEEE Trans. Wireless Commun.*, to be published.

[8] V. V. Chetlur and H. S. Dhillon, "Downlink coverage analysis for a finite 3-D wireless network of unmanned aerial vehicles," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4543–4558, Jul. 2017.

[9] Z. Yang, H. Xu, J. Shi, Y. Pan, and Y. Li, "Power control and resource allocation for multi-cell OFDM networks," in *Proc. Int. Conf. Comput. Commun. Workshops (INFOCOMW)*, San Francisco, CA, USA, Apr. 2016, pp. 891–896.

[10] F. Wang, C. Xu, L. Song, and Z. Han, "Energy-efficient resource allocation for device-to-device underlay communication," *IEEE Trans. Wireless Commun.*, vol. 14, no. 4, pp. 2082–2092, Apr. 2015.