Energy-Efficient UAV enabled Data Collection via Wireless Charging: A Reinforcement Learning Approach

Shu Fu, Yujie Tang, Yuan Wu, Ning Zhang, Huaxi Gu, Chen Chen, and Min Liu

***Abstract*—In this paper, we study the application of unmanned aerial vehicle (UAV) for data collection with wireless charging, which is crucial for providing seamless coverage and improving system performance in the next generation wireless networks. To this end, we propose a reinforcement learning based approach to plan the route of UAV to collect sensor data from sensor devices scattered in the physical environment. Specifically, the physical environment is divided into multiple grids, where one spot for UAV hovering as well as the wireless charging of UAV is located at the center of each grid. Each grid has a spot for the UAV to hover, and moreover, there is a wireless charger at the center of each grid which can provide wireless charging to the UAV when it is hovering in the grid. When the UAV lacks energy, it can be charged by the wireless charger at the spot. By taking into account the collected data amount as well as the energy consumption, we formulate the problem of data collection with UAV as a Markov decision problem, and exploit the Q-learning to find the optimal policy. In particular, we design the reward function considering the energy-efficiency of UAV flight and data collection, based on which Q-table is updated for guiding the route of UAV. Through extensive simulation results, we verify that our proposed reward function can achieve a better performance in terms of the average throughput, delay of data collection, as well as the energy-efficiency of UAV, in comparison with the conventional capacity based reward function.**

***Index Terms*- unmanned aerial vehicle; energy-efficiency; data collection; reinforcement learning; Q-learning; design of reward function.**

I. Introduction

This work was supported in part by the National Key R&D Program of China under Grant 2018YFE0202800; in part by the National Natural Science Foundation of China under Grant 61701054; in part by the Fundamental Research Funds for the Central University under Grant 2020CDJQY-A001. Y. Wu's work is supported in part by Science and Technology Development Fund of Macau SAR under Grants 0060/2019/A1 and 0162/2019/A3, in part by FDCT-MOST Joint Project under Grant 066/2019/AMJ.

Shu Fu is with the School of Microelectronics and Communication Engineering, Chongqing University, Chongqing, P. R. China, 400044. He is also with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, Shaanxi, 710071, P. R. China. (e-mail: shufu@cqu.edu.cn).

Yujie Tang is with the Department of Computer Science and Mathematics, Algoma University, Sault Ste. Marie, ON, Canada (email: yujie.tang@algomau.ca).

Yuan Wu is with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macao, China. He is also with the Department of Computer Information Science, University of Macau (email: yuanwu@um.edu.mo).

Ning Zhang is with the Department of Electrical and Computer Engineering, University of Windsor, ON, N9B 3P4, Canada (email: ning.zhang@ieee.org).

Huaxi Gu is the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, Shaanxi, 710071, P. R. China. (hxgu@xidian.edu.cn).

Chen Chen and Min Liu are with the School of Microelectronics and Communication Engineering, Chongqing University, Chongqing, P. R. China, 400044. (e-mails: c.chen@cqu.edu.cn, liumin@cqu.edu.cn).

Corresponding author: Min Liu.

AS the explosive increase in the number of sensors in physical environments, collecting data from the widely deployed sensors has become a critical issue to Internet of Things (IoT) [1–3] and terrestrial-satellite networks [4] *etc*. In order to collect the data in physical environment, sensor devices are distributed to collect the traffic demand, environment information *etc*., and put these information into data blocks for further processing. However, considering the practical environment constraints, it is infeasible for sensors to directly communicate with base stations (BSs), *e.g.*, in rural or mountain areas. Fortunately, unmanned aerial vehicle (UAV) [1, 5–11] provides an effective approach to collect sensor data, where UAV can act as a mobile base station (BS) in the air.

Several existing works [12–14] have discussed the typical application scenarios of UAV assisted wireless networks. In general, the applications of UAV in data collection can be categorized into on-line UAV relay between BS and sensors and the off-line UAV collection of sensor data. For the former type of applications, UAV will be the mobile relay to enhance the received signal strength from sensors to BS in real time [6, 15, 16]. For the latter type, the delay should be insensitive for sensor data, and sensors can temporarily cache the detected data as files for transmitting to UAV. In particular, the latter type of applications can be leveraged in many scenarios such as geoenvironmental detection, record transaction in IoT, *etc*.

In this work, we focus on the latter one, *i.e.* UAV based off-line data collection. This scenario has been studied from different aspects to improve the performance of data collection. Zhan *et. al.* in [17] jointly optimize the wakeup schedule of sensors and UAV trajectory to minimize the energy consumption of sensors. Gong *et. al.* in [18] study the flight time minimization problem for completing the UAV based data collection mission in an one-dimensional sensor network. In practice, UAV does not have the global channel state information between sensors and UAV. In addition, the computation as well as the battery capacity of UAV require low-complexity solutions without consuming a heavy computational burden. Machine learning has been recognized as an effective method to plan the route of UAV. Some works have explored the performance gain brought by machine learning in UAV assisted wireless networks. Zhao *et. al.* in [19] propose an improved Q-learning method [20–22] to achieve UAV navigation and obstacle avoidance. Wu *et. al.* in [23] propose a direction-aware Q-learning algorithm to locate the illegal radio station by UAV. Bayerlein *et. al.* in [24] study the throughput maximization problem in UAV based data collection. However, the

aforementioned works do not consider the energy-efficiency of UAV, which may lead to a significant power consumption, and therefore affect the flight time for data collection.

Motivated by the above observations, in this paper, we propose an energy-efficient Q-learning approach to plan the route of UAV for data collection. Q-learning can be modeled as a Markov decision process (MDP), which can derive the optimal solution under the probabilistic behavior of a system. Specifically, we consider a geographical area where a wireless sensor network is deployed and sensors continuously broadcast reference signal. Based on the reference signal received power (RSRP), UAV selects one sensor and receives its data in one time-slot. The spots in system have multiple hovering height [25]. UAV can hover over at a specific height of a spot to collect sensor data. Moreover, to prolong the flight time of the UAV, we deploy a wireless charger at the center of each grid, which provides wireless charging to the UAV when it is in short of energy. Different from the traditional Q-learning approach focusing on only the system throughput gain, we consider the energy consumption of UAV flight and propose an energy-efficiency oriented reward function in this paper. Based on the reward function, a Q-table will be iteratively updated to guide the trajectory of UAV while improving its energy-efficiency.

Starting from a specific spot, UAV will continue moving and hovering over the spots for data collection based on RSRP of sensors. In each time-slot, the UAV flies to a spot to collect data from the sensor in the grid with a certain probability such that a large Q value can be obtained. In our design of Q function, we consider both the UAV's energy consumption for hovering and flying, and the delay of flying and wireless charging for powering. As a result, our proposed Q-learning framework is able to provide an energy-efficient solution for the UAV's trajectory optimization.

The remainder of this paper is organized as follows. We review the related studies in Section II. Section III presents the system model and the problem formulations. Section IV presents the Q-learning framework based on the energy-efficiency of UAV for collecting sensor data. The simulation results are demonstrated in Section V. We finally conclude the work and discuss the future directions in Section VI.

II. Related Works

Reinforcement learning has been widely used in wireless communications. Nie *et. al.* in [26] propose an alternative Q-learning theoretical approach to solve the dynamic channel assignment problem in wireless networks. The feature of Q-learning that it does not require the explicit state transition model to solve the Markov decision problem quickly attracts researchers' attentions. This enables Q-learning to adapt the large state space and action set. Yu *et. al.* in [27] use Q-learning to guarantee the traffic QoS for wireless adaptive multimedia. In recent years, there have been many studies exploiting the reinforcement learning in different scenarios, e.g., device-to-device scenario [28] and wireless systems [29], *etc*.

As the wireless networks evolve, the parameters and scenarios become more and more complex, which can lead to the difficulty of the network planning by the traditional optimization technologies such as convex optimization [4] and geometric programming, *etc*. On the other hand, the global optimal solution is not practical in engineering. Reinforcement learning provides an effective method to learn the network actions from the environment. Such a scalability and flexibility of reinforcement learning cater for the complex network, such as 5G and Beyond 5G networks, especially as the terrestrial-satellite system is employed and a more complex network leading to a large amount of states and actions. Raza *et. al.* in [30] proposes a slice admission strategy based on reinforcement learning in 5G network slicing. Zhang *et. al.* in [31] uses the deep reinforcement learning method to provide proactive caching for the multi-view 3D videos in 5G.

In terms of the applications beyond 5G, UAV-BS is a typical scenario. Yin *et. al.* in [32] uses a deterministic policy gradient based reinforcement learning to generate the trajectory for UAV. Challita in [33] uses a dynamic game based deep reinforcement learning to manage the wireless interference in the scenario of cellular-connected UAVs. Liu in [34] uses reinforcement learning to achieve the dynamic movement of multiple UAVs in a wireless network. Dai in [35] studies the deployment of multiple UAVs in a dynamic environment. Hu in [36] proposes an intelligent handover control method in UAV cellular networks by a deep learning method. Li in [37] surveys the artificial intelligence driven spectrum management in various wireless networks such as UAV networks, *etc*.

In terms of energy harvesting, wireless charging can be used for a timely energy supplement in a UAV based system [38]. Su in [39] employs UAV as a wireless charger to provide the energy supplement of energy constrained devices. Chai in [40] proposes an online UAV-assisted wireless caching with wireless charging.

In the existing works, several aspects of the applications of reinforcement learning have been proposed in wireless networks. However, the existing works mostly consider a scenario with a relatively small number of states or actions. Different from the existing works, in this paper, we consider a more practical scenario of the reinforcement learning based data collection via UAV by accounting for the UAV trajectory, hovering height, wireless power charging, *etc*. We also give an in-depth performance evaluation of the proposed algorithm via extensive simulations.

III. System Model and Problem Formulation

A. System Model

We consider a system with N grids as shown in Fig. 1, where J sensors are scatted in each grid. One spot is at the center of each grid. One wireless charger is configured at each spot. UAV should hover over the wireless charger at a specific grid to replenish energy as well as collect sensor data from a certain sensor device in the system. Although wireless power charging can prolong the working-period of UAV, the cost of wireless chargers is an important part of the system cost. This indicates that the number of wireless chargers should be decreased. Hence, we assume that the wireless chargers are configured at each spot to minimize the number of wireless

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051370, IEEE Internet of Things Journal

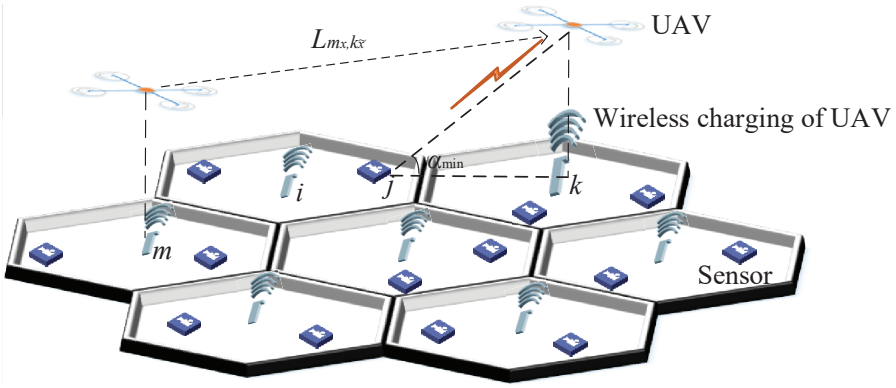


Fig. 1. UAV based data collection.

chargers per grid. At each charging point, UAV has n_{H} predetermined UAV hovering heights which are denoted by set $\boldsymbol{F} = \{F_1, F_2, \ldots, F_{n_{\mathrm{H}}}\}$.

In the system, we assume that the data can only be collected when the UAV is hovering over a wireless charging device. All the sensors in the system possess a file with a length of Z bits for uploading to the UAV. Considering the practical device capability, we assume that the minimum elevation angle from an arbitrary sensor to the UAV is $\alpha_{\min}$. If a sensor has the elevation angle $\alpha < \alpha_{\min}$, it cannot be served by the UAV. This indicates that a larger height of the UAV can have a wider range with more sensors to select by the UAV. On the other hand, sensors continously broadcast reference signals. Based on the strength of the reference signal received power (RSRP) as well as the minimum elevation angle from sensors to the UAV hovering point, UAV can fly to one spot and serve for a specific sensor in the system. When one sensor has completed the data transmission to the UAV, it will stop broadcasting the reference signal. It is noticeable that the UAV may serve for a sensor not located at the grid where the UAV is flying over. As shown in Fig. 1, the UAV serves for the sensor at the grid neighboring to the grid of the UAV stays.

Suppose that the UAV stays at the height of $F_{\tilde{x}}$ above the center of the grid k and serves for the sensor j at the grid i. The transmitting power is assumed as a constant P_0, and the wireless channel gain is denoted by $|h_{ijk_{\tilde{x}}}|^2$. Then, the uplink throughput, $T_{ijk_{\tilde{x}}}$, is

$$T_{ijk_{\tilde{x}}} = \begin{cases} B\log_2(1+\dfrac{P_0|h_{ijk_{\tilde{x}}}|^2}{\sigma^2}), \alpha \geq \alpha_{\min}, \dfrac{P_0|h_{ijk_{\tilde{x}}}|^2}{\sigma^2} \geq \gamma_0; \\ 0, \ \alpha < \alpha_{\min}, \end{cases} \tag{1}$$

where B is the channel bandwidth, and σ^2 denotes the Gaussian white noise power perceived at the UAV. We assume that the wireless bandwidth of sensors is orthogonal with each other. In Eq. (1), γ_0 denotes the threshold of signal to noise ratio (SNR). If either $\alpha < \alpha_{\min}$ or $\frac{P_0|h_{ijk_{\tilde{x}}}|^2}{\sigma^2} < \gamma_0$, UAV cannot serve the sensor when it stays at the grid k from the height of $F_{\tilde{x}}$. In other words, $T_{ijk_{\tilde{x}}} = 0$ in the two cases. Although a higher UAV can have a wider range with more sensors to select by the UAV, the distance between UAV and sensors increases. Eq. (1) indicates that there might be some sensors which cannot be served with a specific hovering height of UAV in the system. In this case, the sensor will not be recorded by the UAV. This process of UAV detection is defined as UAV cruise in Definition 1.

Definition 1 (UAV cruise) at the beginning of data collection, the UAV should first cruise over the N spots from the n_{H} configurations of UAV height in turn. Sensors broadcast reference signals, based on which the UAV can detect the sensors. The UAV records the sensors transmitting reference signals to it with the received SNR above γ_0 and the elevation angle larger than $\alpha_{\min}$. The set of sensors recorded in the UAV is denoted by $\boldsymbol{U}_{m_x}$ for an arbitrary grid m from the height of F_x.

When the UAV flies from the height of F_x at the grid m to the height of $F_{\tilde{x}}$ at the grid k, we denote the distance by $L_{m_x,k_{\tilde{x}}}$ meter (m), and the flight speed of the UAV by V m/s. Then, the corresponding delay of flying can be represented by

$$\tau^f_{m_x,k_{\tilde{x}}} = \frac{L_{m_x,k_{\tilde{x}}}}{V}. \tag{2}$$

We assume that the time-slot of the UAV serving for a specific user is τ_0 second. After the time-slot, the UAV will re-select a spot and hovering height to hover over and serve for a sensor in the system. Additionally, the UAV may be charged for multiple times to receive sensor data. We assume that the wireless charging of the UAV takes τ_w seconds in one time.

In terms of energy consumption, the power consumption of UAV flying is denoted by P_v W/(meter/s), and the power consumption of UAV hovering is denoted by P_s W. Then, the overall energy consumption of the UAV flying at the height of F_x from spot m to the height of $F_{\tilde{x}}$ of the spot k and serving for a specific sensor with one time-slot is

$$E^f_{m_x,k_{\tilde{x}}} = P_v L_{m_x,k_{\tilde{x}}} + P_s\tau_0. \tag{3}$$

Based on Eqs. (1) and (3), the corresponding energy-efficiency of the UAV flying from the height of F_x of spot m to the height of $F_{\tilde{x}}$ at the spot k, and serving for the sensor j in at the grid i ($j \in \boldsymbol{U}_{k_{\tilde{x}}}$) with one time-slot can be given by

$$G^{m_x,k_{\tilde{x}}}_{i,j} = \frac{T_{ijk_{\tilde{x}}}}{P_v L_{m_x,k_{\tilde{x}}} + P_s\tau_0}, j \in \boldsymbol{U}_{k_{\tilde{x}}}. \tag{4}$$

When the objective is to maximize the energy-efficiency while the UAV is collecting data, $G^{m_x,k_{\tilde{x}}}_{i,j}$ should be maximized, where $k_{\tilde{x}}$ is the variable. Similarly, when the objective

is to maximize the throughput while the UAV is collecting data, $T_{ijk_{\widetilde{x}}}$ should be maximized, where $k_{\widetilde{x}}$ is the variable. Since the the global wireless channel gain and the distance between the spots are unavailable in the UAV, both the two system optimization models cannot be directly solved.

Q-learning provides an effective approach for the UAV to learn the wireless channel gains, as well as the obtained throughput and energy consumption under different UAV actions, and generate the optimal UAV route to maximize the energy-efficiency of throughput.

B. Reinforcement Learning Based UAV Route

Q-Learning is a branch of reinforcement learning to deal with the decision of multi-states [41], which is the generalization of MDP. We define the state space of UAV by $\boldsymbol{S}$, action space of UAV by $\boldsymbol{A}$, the reward by $\boldsymbol{G}$, and the probability of the state transforming by $\boldsymbol{P}$. The process of MDP can be denoted by MDP$< \boldsymbol{S}, \boldsymbol{A}, \boldsymbol{G}, \boldsymbol{P} >$ as follows.

- $\boldsymbol{S}$: the state space denotes a specific height F_x above a spot *m* where the UAV is hovering over. After the implementation of UAV cruise by Definition 1, the system has $N \times n_{\mathrm{H}}$ states in $\boldsymbol{S}$ denoting the hovering locations of the UAV.
- $\boldsymbol{A}$: the action space denotes that the UAV flies from the height of F_x of a specific spot *m* to another specific height of $F_{\widetilde{x}}$ of spot k and serves for a specific sensor *j* in $\boldsymbol{U}_{k_{\widetilde{x}}}$ in grid *i*, which is denoted by $a_{ij}^{m_x \to k_{\widetilde{x}}}$. For an arbitrary state, there are $N \times n_{\mathrm{H}} \times N \times J$ available actions in $\boldsymbol{A}$ for the UAV.
- $\boldsymbol{G}$: for an arbitrary action, the UAV can obtain a reward to measure the value of the action. We denote the state of the *t*-th ($t \geq 1$) time of UAV movement by S_t, and the corresponding action by A_t. After the UAV takes the action A_t for the state S_t, it can obtain a reward $G_t(S_t, A_t)$ belonging to the reward space $\boldsymbol{G}$.
- $\boldsymbol{P}$: define $\mathcal{P}[S_{t+1}|S_t]$ in $\boldsymbol{P}$ as the probability of the state S_t transforming to S_{t+1}. Since$\mathcal{P}[S_{t+1}|S_t]$ is independent of all previous states and actions, $\mathcal{P}[S_{t+1}|S_1, S_2, \ldots, S_t] = \mathcal{P}[S_{t+1}|S_t]$. Then, a transition matrix can be obtained to demonstrates the probabilities of transitioning from one state to another.

The objective of Q-learning is to learn from the $\boldsymbol{P}$ in the MDP to find an optimal strategy π^* [41] for maximizing the cumulative sum of all future rewards. Choosing a strategy π, the action a_t can be denoted by $\pi(a_t)$, and the cumulative discount sum of all the future rewards using strategy π is

$$G_\pi = \sum_{t=1}^{T} \delta^{t-1} G_t(S_t, \pi(a_t)), \tag{5}$$

where $0 \leq \delta < 1$ is a discount factor, and T is the number of rewards for accumulation. δ determines the weight of the future reward. Denote the space of all the enabled strategies by Λ. The optimal strategy is

$$\pi^* = \arg_{\pi \in \Lambda} \max r_\pi. \tag{6}$$

We can obtain a value measuring the total reward from this state over time as follows:

$$Q_t(S_t, A_t) = \sum_{k=0}^{\infty} \delta^k G_{t+k+1}(S_{t+k+1}, A_{t+k+1}). \tag{7}$$

By Bellman equation, for a state S_t, the average value over time is calculated by $\widetilde{Q}(S_t)$ as:

$$\begin{aligned} \widetilde{Q}(S_t) &= E[G_t|S_t] \\ &= E[G_{t+1} + \delta G_{t+2} + \delta^2 G_{t+2} + \ldots)|S_t] \\ &= E[G_{t+1} + \delta \widetilde{Q}(S_{t+1})|S_t]. \end{aligned} \tag{8}$$

Based on $\{\widetilde{Q}(S_t)\}$, we can obtain a convergence of Q value to guide the action for each state. The $\{\widetilde{Q}(S_t)\}$ can be dealt as a Q-table, and the process above is called Q-learning.

By Q-learning, a reward function is employed to optimize the action of UAV. In this paper, we use Eq. (4) as our reward function to command UAV taking actions with energy-efficiency awareness. Q-learning iteratively improves the state-action value function by updating Q-table based on the Q function, which is represented by its received reward plus an expectation of total future rewards in its next state. The optimal Q-value function is

$$Q^*(S_t, A_t) = E[G_t(S_t, A_t) + \delta \max_{A_{t+1}} Q^*(S_{t+1}, A_{t+1})]. \tag{9}$$

In particular, the optimal strategy π^* can be obtained by the optimal Q-value function in Eq. (9) as folows.

$$\pi^*(S_t) = \arg_{A_t} \max Q^*(S_t, A_t). \tag{10}$$

The Q values are stored in Q-table and updated iteratively. In practice, UAV approximates the optimal Q-function based on the observations of the environment, and updates the Q value in Q-table by Eq. (11) [41].

$$\begin{aligned} Q(S_t, A_t) \leftarrow &(1 - \psi) Q(S_t, A_t) \\ &+ \psi[G_t(S_t, A_t) + \delta \max_{A_{t+1}} Q^*(S_{t+1}, A_{t+1})]. \end{aligned} \tag{11}$$

The parameter ψ ($0 \leq \psi \leq 1$) denotes the learning rate. Since Q-learning is an iterative algorithm, the Q value function will converge to optimal strategy π^* under a certain condition [42, 43] as follows. In this paper, we set $\psi = 1$, which means that the UAV learns very quickly. The Q function is designed and updated iteratively by Eq. (12) below:

$$Q(S_t, A_t) \leftarrow G_{ij}^{m_x \to k_{\widetilde{x}}} + \delta \times \max_{i',j',k',x'} Q(k, a_{i'j'}^{k_{\widetilde{x}} \to k'_{x'}}), \tag{12}$$

where $A_t = a_{ij}^{m_x \to k_{\widetilde{x}}}$, $S_t = m_x$, and $S_{t+1} = k_{\widetilde{x}}$. Parameter δ in $[0, 1)$ is a discount factor. $\delta \to 1$ makes Q function focusing on the long-term reward, and $\delta \to 0$ makes Q function focusing on the immediate reward of a pair of state and action [43].

When the length of the file Z at each sensor is large enough, the reward of each action can be kept for a relatively long time, where a stable Q-table can be obtained. However, if Z is relatively small, the reward for each action will vary frequently, resulting in a unstable Q-table.. Hence the Q-table works when Z is relatively large [43].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051370, IEEE Internet of Things Journal

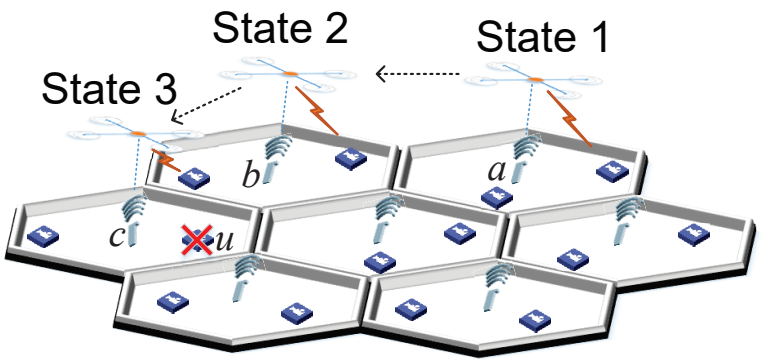


Fig. 2. UAV states and actions.

In practice, from a specific state $S_t = m_x$ at the *t*-th movement, UAV has a probability of $P_a = \xi$ for taking action by

$$A_t = \arg\max_{\tilde{A}_t} Q\left(m_x, \tilde{A}_t\right). \tag{13}$$

By Eq. (13), UAV will serve for the sensor *j* ($j \in \boldsymbol{U}_{k_{\tilde{x}}}$) in the grid *i* from the height of $F_{\tilde{x}}$ above the spot *k*, corresponding to $A_t = a_{ij}^{m_x \to k_{\tilde{x}}}$. On the other hand, UAV has a probability of $P_a = 1 - \xi$ for taking a random or pre-determined action A_t, where UAV flies to a random or pre-determined spot and serves for a random or pre-determined sensor *j* of a grid *i*. This indicates that the UAV has a probability of $P_a = \xi$ to maximize its received reward, yet it also has a probability of $P_a = 1 - \xi$ to explore the received reward of more pairs of actions and states.

IV. Q-Learning Theoretical Data Collection Algorithm by UAV

A. Q-learning based energy-efficient data collection

The *N* spots constitute of the state space for the UAV's hovering and wireless charging. When the system objective is to maximize the energy-efficiency of the UAV while it is collecting the data, $\{G_{i,j}^{m_x,k_{\tilde{x}}}\}$ in Eq. (4) is the reward.

The actions involved in UAV flight can be illustrated by Fig. 2. Regarding each state in the UAV movement, UAV takes a probability of $P_a = \xi$ to decide its movement based on Q-function and a probability of $P_a = 1 - \xi$ to decide its movement based on a random or pre-determined manner. In State 1, UAV hovers over the spot *a* with a specific hovering height and serves for a sensor in grid *a*. In State 2, UAV moves to the spot *b* with a specific hovering height and serves for a specific sensor in grid *b*. In State 3, UAV flies to the spot *c* with a specific hovering height and serves for a specific sensor in the grid *b*. This indicates that UAV can select all sensors in system which meets the minimal elevation $\alpha_{\min}$ and the threshold of SNR γ_0 from an arbitrary spot. When a specific sensor has transmitted all the data in its cache with *Z* bits, the sensor stops transmitting reference signal and is deleted from the set of sensors in the UAV, *e.g.*, the user *u* in the grid *c*.

We denote a random value φ according to a uniform distribution within [0,1]. If $\varphi < \xi$, then the UAV decides its movement A_t based on Q-function in Eq. (13). If $\varphi \geq \xi$, we assume that the UAV decides its movement A_t by a pre-determined manner, where the UAV serves for the sensor with the least amount of data collected in the UAV. In this case, if the amount of data for sensor *j* in grid *i* collected in the UAV is the least, the sensor will be scheduled. Define the amount of collected data in the UAV for an arbitrary sensor $\tilde{j}$ in grid $\tilde{i}$ by $\beta_{\tilde{i}\tilde{j}}$. Then, the index $< i, j >$ of the selected sensor meets

$$< i, j >= \arg\min_{\tilde{i},\tilde{j}} \left(\beta_{\tilde{i}\tilde{j}}\right). \tag{14}$$

After the pair $< i, j >$ has been determined by Eq. (14), the spot *k* and hovering height $F_{\tilde{x}}$ of the UAV meet

$$k_{\tilde{x}} = \arg\max_{\tilde{A}_t} Q\left(S_t, \tilde{A}_t\right) = \arg\max_{k,\tilde{x}} \left(Q\left(m_x, a_{ij}^{m_x \to k_{\tilde{x}}}\right)\right), \tag{15}$$

where m_x is the state of S_t, and $A_t = a_{ij}^{m_x \to k_{\tilde{x}}}$ in the case of $\varphi \geq \xi$. Considering Eqs. (13), (14), and (15), we can determine the UAV action by Eq. (16):

$$A_t = \begin{cases} \arg\max_{\tilde{A}_t} Q\left(m_k, \tilde{A}_t\right), \varphi < \xi, \\ a_{ij}^{m_x \to k_{\tilde{x}}} \text{ based on Eqs. (14), and (15)}, \varphi \geq \xi, \end{cases} \tag{16}$$

where in the case of $\varphi \geq \xi$, the index pair of the served sensor $< i, j >$ is determined by Eq. (14), and the next spot *k* and hovering height $F_{\tilde{x}}$ of UAV is determined by Eq. (15) in the case of $\varphi \geq \xi$.

We assume that the energy consumption for the *t*-th movement of the UAV is denoted by E_t, the corresponding starting spot and hovering height of the UAV by m_x^t, and the next spot as well as the hovering height of the UAV by $k_{\tilde{x}}^t$. Then, E_t can be expressed by

$$E_t = E_{m_x^t k_{\tilde{x}}^t}^f = P_v L_{m_x^t, k_{\tilde{x}}^t} + P_s \tau_0. \tag{17}$$

The overall energy-efficiency *W* of UAV can be calculated at the UAV as:

$$W = \frac{\sum_{1 \leq i \leq N} \sum_{1 \leq j \leq J} \beta_{ij}}{\sum_{t=0}^{\widetilde{T}} E_t}, \tag{18}$$

where $\widetilde{T}$ is the overall number of movement that the UAV takes to collect sensor data in the system. In Eq. (18), only sensor data collected by the UAV is considered, *i.e.*,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051370, IEEE Internet of Things Journal

Algorithm 1 Implementation of QEDU algorithm.

Input:
Initialize the number of UAV movement $t = 0$; Initialize Q-table by an empty table; Initialize reward space $\boldsymbol{G}$ by an empty set, $\boldsymbol{G} = \emptyset$; State space $\boldsymbol{S} = \{1, 2, \ldots, N\}$; Action space $\boldsymbol{A} = \{a_{ij}^{m_x \to k_{\widetilde{x}}}\}$; Initialize $\beta_{ij} = 0$, $\forall i$, and $\forall j$; The length of data for transmission per sensor Z; a small positive value δ; ξ $(0 \le \xi \le 1)$; γ_0; Initial the spot of UAV by $S_0 = m_x$.

Output:
Energy-efficiency W and overall delay D.

1: **Step 1) detecting sensors in system:**
2: Implement the UAV cruise in Definition 1, and obtain $\{\boldsymbol{U}_{k_{\widetilde{x}}}\}$, $\forall i$.
3: UAV flies back to the spot m_x.
4: **Step 2) Q-learning based data collection for QEDU:**
5: **while** $|\beta_{ij} - Z| > \delta$ $(i \in \boldsymbol{S}$ and $j \in \boldsymbol{U}_i)$ **do**
6: Generate a random value φ $(0 \le \varphi \le 1)$;
7: Determine the t-th action A_t by Eq. (16);
8: Update reward function $\boldsymbol{G}$ by $G_{ij}^{m_x \to k_{\widetilde{x}}}$ in Eq. (4);
9: Update $Q(S_t, A_t)$ by Eq. (12);
10: $t + 1 \to t$;
11: $S_t = k_{\widetilde{x}}$.
12: **end while**
13: Calculate energy-efficiency W and overall delay D of UAV.

Algorithm 2 Implementation of QTDU algorithm.

Input:
Initialize the number of UAV movement $t = 0$; Initialize Q-table by an empty table; Initialize reward space $\boldsymbol{G}$ by an empty set, $\boldsymbol{G} = \emptyset$; State space $\boldsymbol{S} = \{1, 2, \ldots, N\}$; Action space $\boldsymbol{A} = \{a_{ij}^{m_x \to k_{\widetilde{x}}}\}$; Initialize $\beta_{ij} = 0$, $\forall i$, and $\forall j$; The length of data for transmission per sensor Z; a small positive value δ; ξ $(0 \le \xi \le 1)$; γ_0; Initial the spot of UAV by $S_0 = m_x$.

Output:
Energy-efficiency W and overall delay D.

1: **Step 1) detecting sensors in system:**
2: Implement UAV cruise in Definition 1, and obtain $\{\boldsymbol{U}_{k_{\widetilde{x}}}\}$.
3: UAV flies back to the spot m_x.
4: **Step 2) Q-learning based data collection for QTDU:**
5: **while** $|\beta_{ij} - Z| > \delta$ $(i \in \boldsymbol{S}$ and $j \in \boldsymbol{U}_i)$ **do**
6: Generate a random value φ $(0 \le \varphi \le 1)$;
7: Determine the t-th action A_t by Eq. (24);
8: Update reward function $\boldsymbol{G}$ by $\widetilde{G}_{ij}^{m_x \to k_{\widetilde{x}}}$ in Eq. (21);
9: Update $Q(S_t, A_t)$ by Eq. (22);
10: $t + 1 \to t$;
11: $S_t = k_{\widetilde{x}}$.
12: **end while**
13: Calculate energy-efficiency W and overall delay D of UAV.

$\sum_{1 \le i \le N} \sum_{1 \le j \le J} \beta_{ij} \le NJZ$. Hence, a smaller γ_0 can improve the amount of data collected in the UAV, corresponding to a stronger capacity of data receiving for the UAV.

We denote the delay of flying in Eq. (2) for the t-th movement by $\tau_{m_t k_t}^f$. The overall delay D of UAV can be denoted as:

$$D = \sum_{t=0}^{\widetilde{T}} \tau_{m_t k_t}^f + \tau_w \times \left\lceil \frac{\sum_{t=0}^{\widetilde{T}} E_t}{E_w} \right\rceil, \tag{19}$$

where E_w is the supplement of energy by wireless charging in one time. The delay of wireless charging is denoted by τ_w for one time. In Eq. (19), the number of wireless charging is denoted by $\left\lceil \frac{\sum_{t=0}^{\widetilde{T}} E_t}{E_w} \right\rceil$. In practice, the rate of wireless charging $\frac{E_w}{\tau_w}$ denotes the capacity of wireless charging at the spots.

The overall data amount collected by UAV is

$$Y = \sum_{1 \le i \le N} \sum_{1 \le j \le J} \beta_{ij}. \tag{20}$$

Then, the Q-learning based energy-efficient data collection by UAV (*i.e.*, QEDU) algorithm can be illustrated in Algorithm 1. By QEDU, UAV can learn an energy-efficient route to collect sensor data. In Line 1, the UAV detects sensors by the UAV cruise in Definition 1. Afterwards, the UAV flies back to the initial spot m_x. In Line 4, Q-learning based data collection by the UAV is implemented in an iterative manner. In Line 6, a random variable φ is used to decide the method generating an action from the current state m. In Line 7, the action is generated by Eq. (16). In Line 8, the reward function is updated by the energy-efficiency involved in the action in Line 7. In Line 9, Q-function can be updated based on the reward function. From Line 10 to Line 11, variables are updated for the next iteration. The iteration will be continued until all the data has been collected from sensors in $\{\boldsymbol{U}_{m_x}\}$ by the UAV as in Line 5.

B. Q-learning based throughput-maximizing data collection

When the system objective is to maximize the throughput of the UAV while it is collecting the data, $\{T_{ijk_{\widetilde{x}}}\}$ in Eq. (1) is the reward. This constructs a Q-learning based throughput-maximizing data collection by UAV (*i.e.*, QTDU) algorithm proposed by Algorithm 2.

In QTDU, the reward function $\widetilde{G}_{ij}^{m_x \to k_{\widetilde{x}}}$ is based on the amount of sensor data collected by UAV flying from spot m with the height of F_x to k with the height of $F_{\widetilde{x}}$, and serving for the sensor j of $\boldsymbol{U}_{k_{\widetilde{x}}}$ in the grid i with one time-slot as

$$\widetilde{G}_{ij}^{m_x \to k_{\widetilde{x}}} = T_{ijk_{\widetilde{x}}}. \tag{21}$$

Like Eq. (12), we can obtain the Q function of QTDU, denoted by $\widetilde{Q}(S_t, A_t)$, in Eq. (22).

$$\widetilde{Q}(S_t, A_t) \leftarrow \widetilde{G}_{ij}^{m_x \to k_{\widetilde{x}}} + \gamma \times \max_{i', j', k', x'} \widetilde{Q}(k, a_{i'j'}^{k_{\widetilde{x}} \to k'_{x'}}). \tag{22}$$

In the case that the generated random value φ meeting $\varphi \ge \xi$ in Line 6 of QTDU algorithm, the spot and the height of the UAV in Eq. (15) can be re-written as $k_{\widetilde{x}}$ in Eq. (23).

$$k_{\widetilde{x}} = \arg\max_{\tilde{A}_t} \widetilde{Q}\left(S_t, \tilde{A}_t\right) = \arg\max_{k, \widetilde{x}} \left(\widetilde{Q}\left(m_x, a_{ij}^{m_x \to k_{\widetilde{x}}}\right)\right), \tag{23}$$

where m_x is the state of S_t, $< i, j >$ is determined by Eq. (14). and $A_t = a_{ij}^{m_x \to k_{\widetilde{x}}}$.

Considering Eqs. (13), (14), and (23), we can determine the UAV action A_t of QTDU as:

$$A_t = \begin{cases} \arg\max_{\tilde{A}_t} \widetilde{Q}\left(m_x, \tilde{A}_t\right), \varphi < \xi, \\ a_{ij}^{m_x \to k_{\widetilde{x}}} \text{ based on Eqs. (14), and (15)}, \varphi \ge \xi. \end{cases} \tag{24}$$

The outputs of QTDU are energy-efficiency W and overall delay D of UAV as in Line 13 of QTDU algorithm, which will be compared with the performance of QEDU.

As discussed above, QEDU and QTDU can generate the route of the UAV with different objective functions. QEDU can generate the route with energy-efficiency awareness, and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051370, IEEE Internet of Things Journal

QTDU aims to maximize UAV's receiving sensor data at each spot. In the future work, we will further formulate the system by deep learning to adapt to a more complex environment [44].

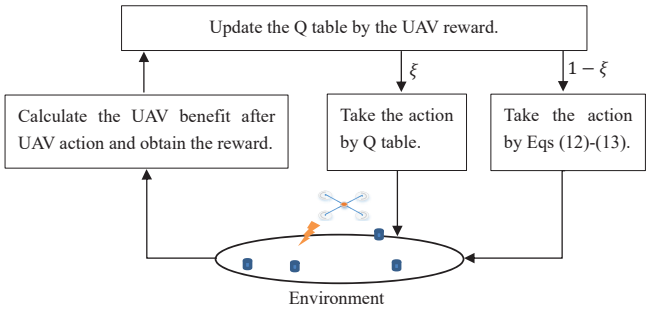
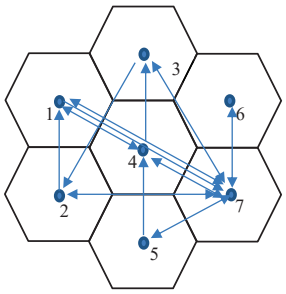


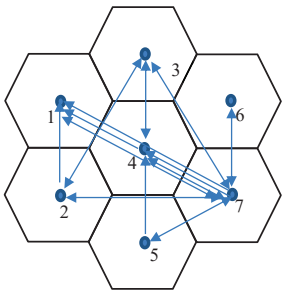
Fig. 3. The implementation of the Q-learning based data collection by UAV.

By exploiting the two algorithms, we summarize the process of Q-learning based data collection by UAV as in Fig. 3, where the UAV first exchanges information with the wireless environment to obtain the set of $\{\boldsymbol{U}_{m_x}\}$. Then, the Q-table is initialized as an empty table and stored in the UAV. Starting from an arbitrary location with a random action, the UAV can obtain a reward based on either the energy-efficiency or the amount of the collected data, based on which the Q-table can be updated. To determine the next action of the UAV, two methods are used. The first one is based on the Q-table with the probability of ξ to maximize the reward. The shortcoming of this method is that the potential rewards brought by the other actions may not be explored, and the current optimal reward based on the Q-table may be not the optimal. The second one is that the UAV serves the sensor with the least amount of data collected in the UAV cache with the probability of $1-\xi$. Since all the data in $\{\boldsymbol{U}_{m_x}\}$ need to be collected, such a method can extend the Q-table and provide more actions for the UAV to obtain the potential larger rewards. This can also speed up the Q-table updating in practice. The disadvantage behind the second method is that the frequent implementation of this method, poor rewards may be obtained due to the long distance of the UAV flying. Hence, an appropriate tradeoff between the two methods should be determined with the parameter ξ. After the action, the state of the UAV will be updated, and the above process is implemented in an iterative manner.

In Fig. 4, we show the UAV trajectory when $Z = 5 \times 10^3$. The UAV trajectory for QEDU and QTDU are given in Fig. 4 (a) and (b), respectively. By the exploring when $\varphi \geq \xi$, the UAV trajectory will be gradually extended. However, the different rewards lead to different UAV actions. We give the specific performance parameters involved in the Q-learning algorithms in Fig. 4 (c). The action of QEDU is determined by maximizing the energy efficiency, which can also decrease the total system delay. The action of QTDU is determined by maximizing the amount of the data collection, where the energy consumption and delay are not considered. We can find that the overall flying distance of UAV with QEDU is less than that with QTDU. This is because that QEDU considers the energy consumption in its action decision. Thanks to the time saving of QEDU to improve the system energy-efficiency, QEDU outperforms QTDU in terms of both the system delay and energy-efficiency.



(a) The UAV flight by QEDU.



(b) The UAV flight by QTDU.

Parameter	QEDU	QTDU
The overall distance of UAV flying (km)	233.2	344.55
The time for data collection (hour)	0.09	0.15
The energy efficiency (bps/J)	8.3×10^8	2.9×10^8

(c) The performance of the Q-learning based algorithms.

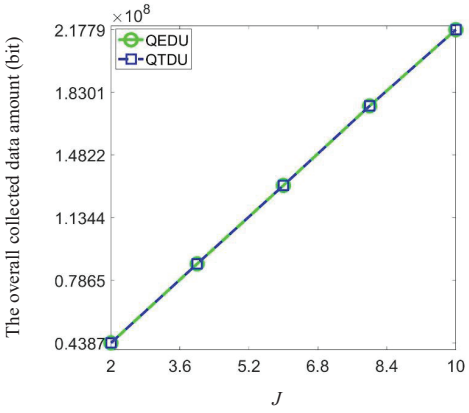
Fig. 4. The performance of algorithms by Q-learning.

V. NUMERICAL RESULTS

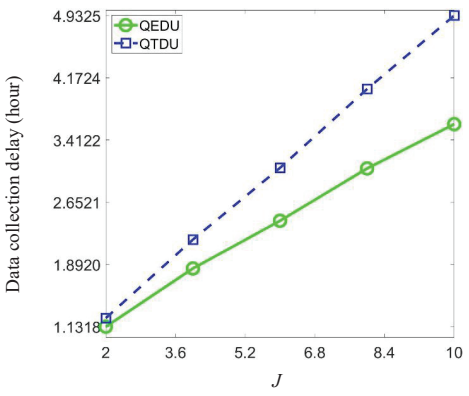
TABLE I
SIMULATION PARAMETERS

Parameter	Value
δ	0.5
N	7
J	4
B	50 MHz
The cell radius of grid L_0	500 meter
The minimal height of UAV ζ	10 meter
Z	2×10^6 bits
τ_w	10 s
τ_0	10^{-3} s
γ_0	100
E_w	10^{-6} J
B	50 MHz
V	20 m/s
$\alpha_{\min}$	$\arctan(\beta/L_0)$
P_0	2×10^{-3} W
P_v	2×10^{-8} W/(m/s)
P_s	2×10^{-10} W
ξ	0.5
n_H	2
Gaussian White noise power spectral density	-174 dBm/Hz

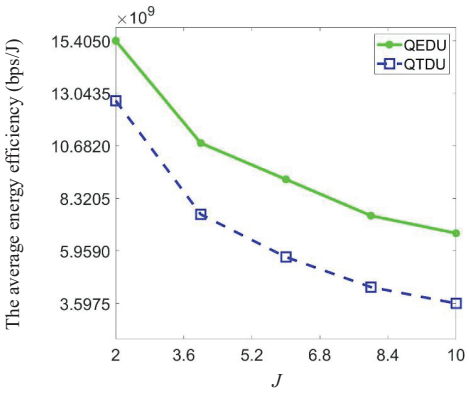
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051370, IEEE Internet of Things Journal



(a) The overall collected data amount *v.s.* *J*.

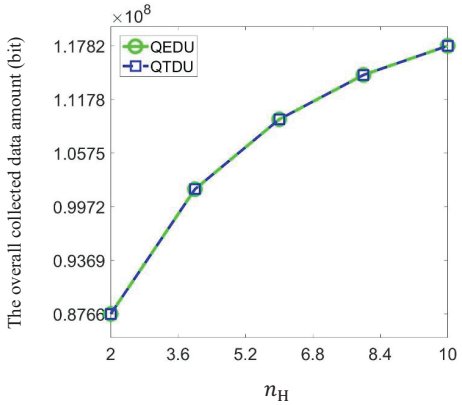


(b) The data collection delay *v.s.* *J*.

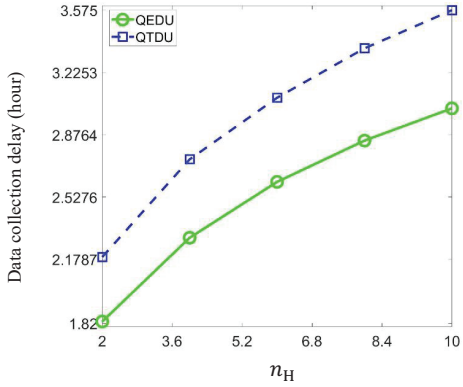


(c) The average energy efficiency *v.s.* *J*.

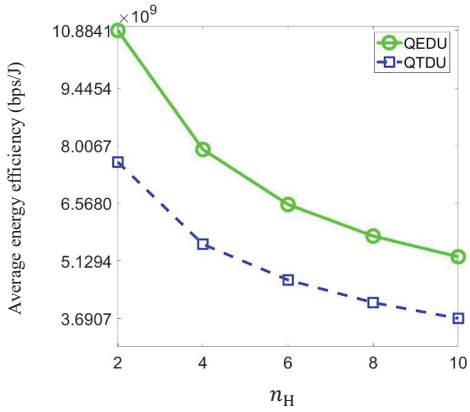
Fig. 5. The performance of algorithms with *J*.



(a) The overall collected data amount *v.s.* n_{H}.



(b) The data collection delay *v.s.* n_{H}.



(c) The average energy efficiency *v.s.* n_{H}.

Fig. 6. The performance of algorithms with n_H.

In the simulation section, we employ the path loss model as

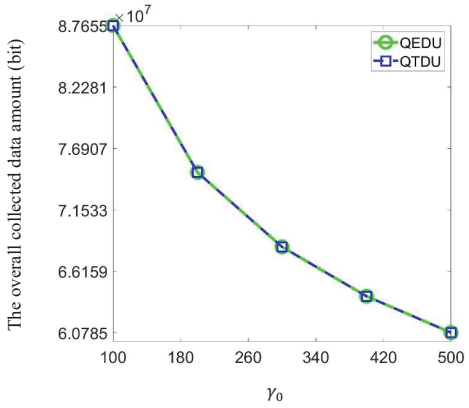
$$\mathfrak{L} = 92.44 + 20 \times \log_{10}(L) + 20 \times \log_{10}(f) \text{ dB},$$

where f is the system operating frequency and $f = 2$ GHz. The unit of the distance between UAV and sensors, L, is kilometer (km), and the unit of the frequency is GHz in the path loss model. We consider the fast fading as complex Gaussian distribution $\mathcal{CN}(0, 1 \text{ dB})$, and the shadow model as log-normal distribution $\mathcal{C}(0, 5 \text{ dB})$. Unless otherwise specified, the simulation parameters are given in Table I. The minimal height of UAV is denoted by β, and the available height of UAV is determined by $\boldsymbol{F} = [1, 2, \ldots, n_{\mathrm{H}}] \times \zeta$.

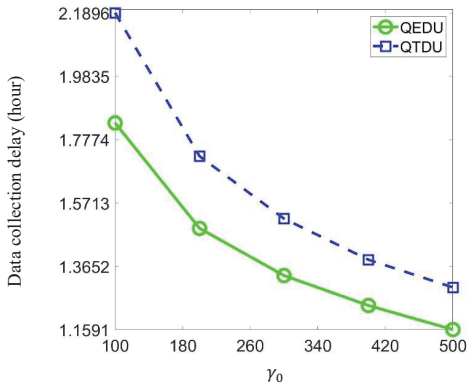
For comparison, we take the traditional QTDU algorithm as the compared algorithm to QEDU algorithm.

In Fig. 5, the performance of algorithms with *J* is illustrated. In Fig. 5 (a), we compare the performance of QEDU and QTDU in terms of data amount collected by UAV as in Eq. (20). We can find the data amount collected by UAV is the same under the two algorithms because the UAV will continue to work until all the available data under the constraints of $\alpha_{\min}$ and γ_0 has been collected. The same performance can also be found in Fig. 6 (a), Fig. 7 (a), and Fig. 8 (a). As *J* increases, the performance increases due to the larger number of sensors

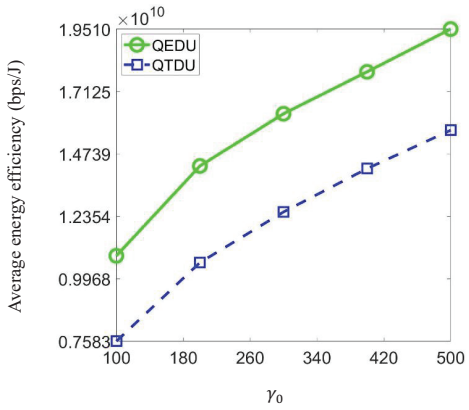
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051370, IEEE Internet of Things Journal



(a) The overall collected data amount *v.s.* γ_0.

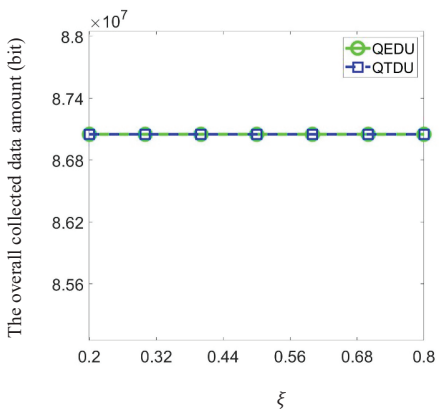


(b) The data collection delay *v.s.* γ_0.

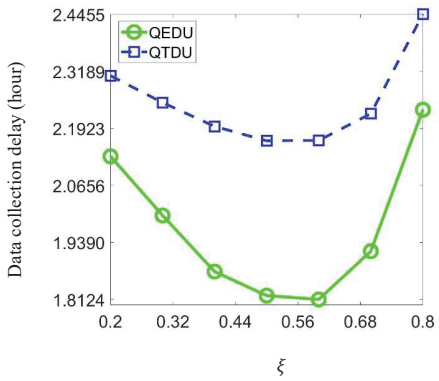


(c) The average energy efficiency *v.s.* γ_0.

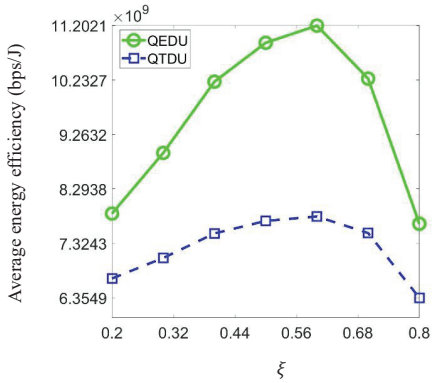
Fig. 7. The performance of algorithms with γ_0.



(a) The overall collected data amount *v.s.* ξ.



(b) The data collection delay *v.s.* ξ.



(c) The average energy efficiency *v.s.* ξ.

Fig. 8. The performance of algorithms with ξ.

providing the larger amount of user data. In Fig. 5 (b), we can find that QEDU outperforms QTDU because on one hand, the energy-efficiency aware UAV movement can largely decrease the system delay. On the other hand, since the received amount of sensor data is invariant under the two algorithms, QEDU can always outperforms QTDU. By Fig. 5 (b), we find that the larger number of sensors leads to the higher data collection delay due to the larger flying distance of UAV. In Fig. 5 (c), we can confirm that QEDU can save more energy than QTDU. As J increases, the average energy-efficiency monotonously decreases due to the increased collected data amount is less than the increased energy consumption.

In Fig. 6, we study the algorithms performance with the different configurations of n_{H}. As n_{H} increases, the available number of the UAV hovering height increases. When n_{H} increases, the number of sensors covered by it increases. On the other hand, as UAV height increases, the number of sensors meeting the constraint of γ_0 decreases. This indicates that, the performance gain brought by increasing n_{H} is decreased when n_{H} is sufficiently large as shown in Fig. 4. Similar to the cases in Fig. 3, In Fig.4 (a), the collected data amount increases due to the increased number of sensors covered by the UAV. In Fig. 6 (b), as n_{H} increases, the data collection delay increases because the UAV should spend more time for learning the UAV

flight to collect the data. In Fig. 6 (c), as n_{H} increases, the energy-efficiency decreases under the configuration in Table I. It is notably that if P_v and P_s is decreased by the improved UAV technologies, the energy-efficiency can also increase as n_{H} increases. We do not provide these simulation results under the small P_v and P_s due to the page save.

In Fig. 7, we show the algorithms performance under different configurations of γ_0. It is obviously that when γ_0 increases, the number of sensors with SNR satisfying γ_0 decreases. This leads to the decrease of the collected data by the UAV as in Fig. 7 (a). By the observation in Fig. 7 (b), as γ_0 increases the gap between the performance of QEDU and QTDU narrows because that the number of sensors with SNR meeting γ_0 decreases, providing the less flexibility for QEDU to cut down the delay by the energy-efficient movement of UAV. In Fig. 7 (c), we find that the average energy-efficiency of QEDU outperforms QTDU. This is because the UAV movement of QEDU employs the energy-efficient reward function in Eq. (4). The energy-efficiency increases as γ_0 increases due to the decreased UAV working time for the data collection.

In Fig. 8, the impact of ξ on the Q-learning performance is illustrated. As in Fig. 8 (a), the amount of data collection remains as ξ changes, because the amount of data collection only depends on the values of $\alpha_{\min}$ and γ_0. As in Fig. 8 (b), as ξ increases, the data collection delay first falls and then achieves the minimal value. As ξ further increases, the data collection delay increases. By Fig. 8 (b), we can find the optimal ξ to minimize the system delay. The reason behind the above changes of the algorithms performance is that a more frequent UAV movement based on the Q-function can collect data with either energy-efficient awareness by QEDU or throughput awareness by QTDU. However, the effective guideline via the Q-table is limited due to that the Q-table is updated and extended slowly. On the other hand, a more frequent UAV movement based on the data amount collected in UAV cache can effectively updated the Q-table more faster. However, the UAC flying actions are more frequently motivated without the energy-efficient awareness and throughput awareness, which may increase the flying time of UAV. In Fig. 8 (c), as ξ increases, the energy-efficiency first increases and then achieves the maximal value. As ξ further increases, the energy-efficiency decreases. This is because that the energy consumption and delay are first decreased due to the Q-table guideline and then increased by the more slowly updated Q-table.

VI. Conclusion

In this paper, we have studied Q-Learning based energy-efficient data collection by UAV. We have proposed a framework of a UAV assisted wireless sensor network to collect sensor data. By analyzing the states and actions of the UAV, we formulated the Q-learning based mechanism to collect sensor data while improving the energy-efficiency. Under different scenarios, we have evaluated the performance of the proposed learning mechanism and verified its effectiveness by extensive simulations. In our future work, we will further consider the scenario of multiple UAVs and investigate the cooperation of multi-UAV for efficient data collection.

References

[1] S. Fu, Y. Tang, N. Zhang, L. Zhao, S. Wu, and X. Jian, "Joint unmanned aerial vehicle (UAV) deployment and power control for internet of things networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4367–4378, 2020.

[2] G. Li, S. Peng, C. Wang, J. Niu, and Y. Yuan, "An energy-efficient data collection scheme using denoising autoencoder in wireless sensor networks," *IEEE Tsinghua Science and Technology*, vol. 24, no. 1, pp. 86–96, 2019.

[3] L. Hu, H. Wen, B. Wu, F. Pan, R. Liao, H. Song, J. Tang, and X. Wang, "Cooperative jamming for physical layer security enhancement in internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 219–228, 2018.

[4] S. Fu, J. Gao, and L. Zhao, "Integrated resource management for terrestrial-satellite systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3256–3266, 2020.

[5] H. Baek and J. Lim, "Design of future UAV-relay tactical data link for reliable UAV control and situational awareness," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 144–150, 2018.

[6] S. Fu, L. Zhao, Z. Su, and X. Jian, "UAV based relay for wireless sensor networks in 5G systems," *MDPI Sensors*, vol. 18, no. 8, pp. 1–11, 2018.

[7] B. Ji, Y. Li, B. Zhou, C. Li, K. Song, and H. Wen, "Performance analysis of UAV relay assisted IoT communication network enhanced with energy harvesting," *IEEE Access*, vol. 7, pp. 38 738–38 747, 2019.

[8] J. Li, H. Zhao, H. Wang, F. Gu, J. Wei, H. Yin, and B. Ren, "Joint optimization on trajectory, altitude, velocity and link scheduling for minimum mission time in UAV-aided data collection," *IEEE Internet of Things Journal , early access*, pp. 1–12, 2020.

[9] G. Wu, "UAV-based interference source localization: A multimodal Q-learning approach," *IEEE Access*, vol. 7, pp. 137 982–137 991, 2019.

[10] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Communication Magazine*, vol. 55, no. 7, pp. 101–109, 2017.

[11] H. Wu, X. Tao, N. Zhang, and X. Shen, "Cooperative UAV cluster assisted terrestrial cellular networks for ubiquitous coverage," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 2045–2058, 2018.

[12] M. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao, and X. Shen, "Energy-efficient UAV-assisted mobile edge computing: resource allocation and trajectory optimization," *IEEE Transactions on Vehicular Technology, revision requested*, pp. 1–12, 2019.

[13] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2018.

[14] I. Jawhar, N. Mohamed, and J. Al-Jaroodi, "UAV-based data communication in wireless sensor networks: Models and strategies," in *Proceedings of IEEE Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 1–8.

[15] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for UAV-enabled mobile relaying systems," *IEEE Transactions on Communications*, vol. 64, no. 12, pp. 4983–4996, 2016.

[16] G. Zhang, H. Yan, Y. Zeng, M. Cui, and Y. Liu, "Trajectory optimization and power allocation for multi-hop UAV relaying communications," *IEEE Access*, vol. 6, pp. 48 566–48 576, 2018.

[17] C. Zhan, Y. Zeng, and R. Zhang, "Energy-efficient data collection in UAV enabled wireless sensor network," *IEEE Wireless Communications Letters*, vol. 7, no. 3, pp. 328–331, 2018.

[18] J. Gong, T.-H. Chang, C. Shen, and X. Chen, "Flight time minimization of UAV for data collection over wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 1942–1954, 2018.

[19] Y. Zhao, Z. Zheng, X. Zhang, and Y. Liu, "Q learning algorithm based uav path learning and obstacle avoidence approach," in *Proceedings of IEEE Chinese Control Conference (CCC)*, 2017, pp. 1–6.

[20] J. Li, T. Chai, F. L. Lewis, J. Fan, Z. Ding, and J. Ding, "Off-policy Q-learning: Set-point design for optimizing dual-rate rougher flotation operational processes," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4092–4102, 2018.

[21] J. Zhu, Y. Song, D. Jiang, and H. Song, "A new deep-Q-learning-based transmission scheduling mechanism for the cognitive internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2375–2385, 2018.

[22] J. Duan, H. Xu, and W. Liu, "Q-learning-based damping control of wide-area power systems under cyber uncertainties," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6408–6418, 2018.

[23] S. Wu, "Illegal radio station localization with UAV-based Q-learning,"

IEEE China Communications, vol. 15, no. 12, pp. 122–131, 2018.

[24] H. Bayerlein, R. Gangula, and D. Gesbert, "Learning to rest: A Q-learning approach to flying base station trajectory design with landing spots," in *Proceedings of IEEE Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 724–728.

[25] S. Suman, S. Kumar, and S. De, "Impact of hovering inaccuracy on UAV-aided RFET," *IEEE Communications Letters*, vol. 23, no. 12, pp. 2362–2366, 2019.

[26] J. Nie and S. Haykin, "A Q-learning-based dynamic channel assignment technique for mobile communication systems," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1676–1687, 1999.

[27] F. Yu, V. Wong, and V. Leung, "Efficient QoS provisioning for adaptive multimedia in mobile communication networks by reinforcement learning," in *Proceedings of IEEE International Conference on Broadband Networks*, 2004, pp. 579–588.

[28] A. Asheralieva and Y. Miyanaga, "An autonomous learning-based algorithm for joint channel and power level selection by D2D pairs in heterogeneous cellular networks," *IEEE Transactions on Communications*, vol. 64, no. 9, pp. 3996–4012, 2016.

[29] N. Morozs, T. Clarke, and D. Grace, "Distributed heuristically accelerated Q-learning for robust cognitive spectrum management in LTE cellular systems," *IEEE Transactions on Mobile Computing*, vol. 15, no. 4, pp. 817–825, 2016.

[30] M. R. Raza, C. Natalino, P. Ohlen, L. Wosinska, and P. Monti, "Reinforcement learning for slicing in a 5G flexible RAN," *IEEE Journal of Lightwave Technology*, vol. 37, no. 20, pp. 5161–5169, 2019.

[31] Z. Zhang, Y. Yang, M. Hua, C. Li, Y. Huang, and L. Yang, "Proactive caching for vehicular multi-view 3D video streaming via deep reinforcement learning," *IEEE Transactions on Wireless Communications*, vol. 18, no. 5, pp. 2693–2706, 2019.

[32] S. Yin, S. Zhao, Y. Zhao, and F. R. Yu, "Intelligent trajectory design in UAV-aided communications with reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8227–8231, 2019.

[33] U. Challita, W. Saad, and C. Bettstetter, "Interference management for cellular-connected UAVs: A deep reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2125–2140, 2019.

[34] X. Liu, Y. Liu, and Y. Chen, "Reinforcement learning in multiple-uav networks: Deployment and movement design," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8036–8049, 2019.

[35] H. Dai, H. Zhang, C. Li, and B. Wang, "Efficient deployment of multiple UAVs for IoT communication in dynamic environment," *IEEE China Communications*, vol. 17, no. 1, pp. 89–103, 2020.

[36] B. Hu, H. Yang, L. Wang, and S. Chen, "A trajectory prediction based intelligent handover control method in UAV cellular networks," *IEEE China Communications*, vol. 16, no. 1, pp. 1–14, 2019.

[37] Z. Li, Z. Ding, J. Shi, W. Saad, and L. Yang, "Guest editorial: Artificial intelligence (AI)-driven spectrum management," *IEEE China Communications*, vol. 17, no. 2, pp. iii–v, 2020.

[38] B. Galkin, J. Kibilda, and L. A. DaSilva, "Uavs as mobile infrastructure: Addressing battery lifetime," *IEEE Communications Magazine*, vol. 57, no. 6, pp. 132–137, 2019.

[39] C. Su, F. Ye, L. Wang, L. Wang, Y. Tian, and Z. Han, "UAV-assisted wireless charging for energy-constrained IoT devices using dynamic matching," *IEEE Internet of Things Journal , early access*, pp. 1–12, 2020.

[40] S. Chai and V. K. N. Lau, "Online trajectory and radio resource optimization of cache-enabled UAV wireless networks with content and energy recharging," *IEEE Transactions on Signal Processing*, vol. 68, pp. 1286–1299, 2020.

[41] D. Ebrahimi, S. Sharafeddine, P.-H. Ho, and C. Assi, "Autonomous UAV trajectory for localizing ground objects: A reinforcement learning approach," *IEEE Transactions on Mobile Computing, early access*, pp. 1–13, 2020.

[42] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[43] R. S. Sutton and A. G. Barto, *Introduction to reinforcement learning, 2 nd ed.*, 2018.

[44] N. Ye, X.-M. Li, H. Yu, L. Zhao, W. Liu, and X. Hou, "DeepNOMA: A unified framework for NOMA using deep multi-task learning," *IEEE Transactions on Wireless Communications, early access*, pp. 1–16, 2020.