Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2019.DOI

Adaptive UAV-Trajectory Optimization Under Quality of Service Constraints: A Model-Free Solution

JINGJING CUI[1], (Member, IEEE), ZHIGUO DING[2], (FELLOW, IEEE), YANSHA DENG[3], (MEMBER, IEEE), ARUMUGAM NALLANATHAN[4], (FELLOW, IEEE), AND LAJOS HANZO[1], (FELLOW, IEEE),

[1]School of Electronics and Computer Science, University of Southampton, SO17 1BJ, Southampton (UK) (e-mail: jingj.cui@soton.ac.uk and lh@ecs.soton.ac.uk)
[2]School of Electrical and Electronic Engineering, University of Manchester, Manchester M13 9PL, UK (e-mail: zhiguo.ding@manchester.ac.uk)
[3] Department of Informatics, Kings College London, UK (Email: yansha.deng@kcl.ac.uk).
[4]School of Electronic Engineering and Computer Science, Queen Mary University of London, UK (email:a.nallanathan@qmul.ac.uk)

Corresponding author: Lajos Hanzo (e-mail: lh@ecs.soton.ac.uk).

Part of this work was presented in IEEE Global Commun. Conf. (GLOBECOM), Waikoloa, HI, USA, 2019. L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/Noo4558/1, EP/PO34284/1, COALESCE, of the Royal Society's Global Challenges Research Fund Grant as well as of the European Research Council's Advanced Fellow Grant QuantCom.

ABSTRACT Unmanned aerial vehicles (UAVs) with the potential of providing reliable high-rate connectivity, are becoming a promising component of future wireless networks. A UAV collects data from a set of randomly distributed sensors, where both the locations of these sensors and their data volume to be transmitted are unknown to the UAV. In order to assist the UAV in finding the optimal motion trajectory in the face of the uncertainty without the above knowledge whilst aiming for maximizing the cumulative collected data, we formulate a reinforcement learning problem by modelling the motion-trajectory as a Markov decision process with the UAV acting as the learning agent. Then, we propose a pair of novel trajectory optimization algorithms based on stochastic modelling and reinforcement learning, which allows the UAV to optimize its flight trajectory without the need for system identification. More specifically, by dividing the considered region into small tiles, we conceive state-action-reward-state-action (Sarsa) and Q-learning based UAV-trajectory optimization algorithms (i.e., SUTOA and QUTOA) aiming to maximize the cumulative data collected during the finite flight-time. Our simulation results demonstrate that both of the proposed approaches are capable of finding an optimal trajectory under the flight-time constraint. The preference for QUTOA vs. SUTOA depends on the relative position of the start and the end points of the UAVs.

INDEX TERMS Reinforcement learning, sensor data collection, trajectory optimization, UAV communications

I. INTRODUCTION

In the emerging Internet of Everything (IoE), future networks are expected to autonomously determine the connection of people, processes and things. The exciting applications of unmanned aerial vehicles (UAVs) or drones have drawn considerable attention from academia, industry and regulatory bodies, for expanding the attainable communication coverage and offering on-demand connectivity [1]. Given the versatility and manoeuvrability of UAVs, artificial intelligence aided smart UAV-assisted solutions are capable of enhancing next-generation wireless networks. Hence a range of professional and civil applications of UAVs have been envisioned, including parcel delivery, communications and media, inspection of critical infrastructure, communication relaying, search-and-rescue operations, and surveillance, among others [2], [3].

Due to the limited on-board battery, it is a pressing practical challenge to increase the UAVs' flight time. For example, some off-the-shelf UAVs have a recharge-duration of less than 20 minutes, and a flight range of about 15 miles [4]. Sophisticated laser-charging might come to rescue based on the laser-beam characteristics of monochromaticity and directionality [5]. Furthermore, with imperfect information

of the environment and the communication dynamics of UAV networks, reinforcement learning (RL) becomes a promising technique of improving the control of UAV networks for enhancing their communication qualities, on-demand deployments and trajectory optimization.

A. PRIOR WORKS

In contrast to terrestrial communications, the communication between UAVs and ground devices is generally dominated by line-of-sight (LoS) channels [6], which are beneficial for implementing reliable communications between the UAVs and ground devices. One of the key applications of UAVs in wireless communications is wide-area data collection from geographically dispersed ground devices such as sensors, ground users or ground base stations [7]. Moreover, in UAV aided offloading scenarios [8], [9], the UAV acts as a flying BS or edge server for reducing the tele-traffic. As a result, in addition to conventional wireless resource management, the design of UAV enabled communication networks critically hinges on trajectory optimization due to the mobility of UAVs. By constructing the continuous trajectory as a set of discrete waypoints as well as the UAV speed, Zeng et al. [10] proposed a sequential virtual base station (BS) placement approach for minimizing the mission completion time in a UAV-enabled multicasting system. Furthermore, in [11], the speed control and the data scheduling of UAV enabled sensor networks were investigated and a heuristic algorithm was developed for minimizing the energy consumption. In [12], a dynamic programming based UAV flight-time minimization problem was solved for a one-dimensional sensor network by serving nodes located in a straight line. With the goal of exploiting cloud-like computing functionalities, a successive convex approximation (SCA) based approach for bit allocation between communication and computing as well as path planning of UAV based cloudet was developed in [13] for maximizing the energy efficiency. Furthermore, with successive convex approximation (SCA) techniques, an intelligent reflecting surface aided UAV communication system was investigated in [14] by designing the beamforming and trajectory of the UAV. In [15], a SCA based power and trajectory optimization approach was proposed for the an UAV-aided secure communication network.

If accurate models of UAV networks and of their flight dynamics are available, then the trajectory optimization may be carried out by exploiting standard optimization techniques, such as those in [10]–[13], [16], [17] regardless of the specific objective function (OF) used. However, accurate network models are hard to construct. Hence, there is a need to conceive model-free machine learning techniques to control UAVs in support of wireless services. The ability of machine learning to exploit past experience may be beneficially invoked for formulating an autonomous control policy for UAVs in a timely and flexible manner [18]. In [19], the authors proposed a multi-agent reinforcement learning framework for optimizing multi-UAV networks, where the associated resource allocation scheme was designed by maximizing the systems' long-term reward. Pearre and Brown [20] used a policy gradient method for trajectory optimization by learning the optimal waypoints for minimizing the total traveling distance of the UAV. In [21], the UAV utilized both conventional and deep Q-learning for optimizing the trajectory in order to maximize the sum rate during its flight-time by assuming that the UAV knew nothing about the environment. By modelling the data ferrying task as a Markov Decision Process (MDP), a standard temporal difference (TD) learning technique based on state value functions was proposed in [22] for minimizing the average packet delay by finding an optimal routing policy. Given the promising benefits of UAVs in communication networks, application-driven intelligent trajectory control strategies and resource allocation designs of UAV networks have attracted substantial attention [23], [24]. In [25], a deep reinforcement learning approach based on echo state networks was developed to design the UAV's path with the goal of minimizing the system's total interference in UAV-aided cellular networks. In terms of the network having very large populations of UAVs, an online trajectory optimization approach based on federated learning and mean-field games was proposed in [26]. Moreover, some applications of reinforcement learning in UAV aided networks can be found in [27]–[30]. For instance, an multi-UAV Q-learning approach is proposed for a coordinate UAV network in [27] and the application of reinforcement learning for a cellular UAV network was introduced in in terms of protocol design, trajectory control, and resource management [29].

B. MOTIVATION AND CONTRIBUTIONS

As discussed above, machine learning algorithms provide promising solutions for UAVs by autonomously learning their task-centered control policies in diverse wireless networks [20]–[22], [25], [31]. However, most of these contribution focus on developing standard optimization techniques for the performance analysis of UAV networks under the idealized simplifying assumption of having perfect environmental information available for the UAV. Although some deep learning methods were exploited in recent UAV-enabled wireless networks [25] and [31], an accurate learning model still remains hard to attain due to the dynamics and uncertainties in the environment. On the other hand, reinforcement learning has been used for the control of UAVs in diverse applications [21], [22] by harnessing the UAV as an autonomous agent of learning the optimal policy from its environment. However, given the challenges in modelling UAV-aided communications, there is a paucity of literature on their model-free trajectory optimization.

Hence we fill this gap by developing a reinforcement learning based trajectory optimization framework for UAV-aided data collection, where the transmission time corresponding to the amount of data is considered in the light of the limited on-board energy. In the compact conference paper [32], we introduced a Q-learning based trajectory design for the UAV network. In this article, we extend [32] by introducing the

Sarsa learning algorithm and analyse its pros and cons. More specifically, we consider a wireless network interrogating a set of randomly distributed sensors such as environmental sensors storing data in their memory. Then, the UAV collects the data when flying over the region that sensors are deployed. Explicitly, a bold comparison to different approaches conceived for the UAV's trajectory optimization is provided in Table 1. Based on the proposed framework, our main contributions are as follows:

- We maximize the cumulative data volume of the UAV collected from the sensors by optimizing the flight trajectory between a pair of the start point and the destination subject to the constraint of the limited flight time as well as the received signal quality. Due to the dynamics and uncertainties of the environment, the network information are assume to be unavailable to the UAV. Therefore, the optimization problem formulated is challenging to solve using standard optimization tools since an accurate system model is unknown to the UAV. To tackle these challenges, we develop novel reinforcement learning based trajectory optimization algorithms.
- We transform the trajectory optimization formulated into a multi-period decision problem having a finite number of states and actions by partitioning the region considered into small tiles. Specifically, we model the flight-time-limited trajectory optimization problem formulated as a Markov decision process (MDP), where the tiles represent the state space and the actions correspond to the movements of the UAVs over the tiles. Furthermore, the sensors having finite data volumes for transmission are assumed to be located at the grid points. As the UAV will collect the data when it flies over the sensors, the volume of the data transmitted from a sensor can be treated as the reward attained by the UAV from its communications with that specific sensor. As a result, the problem formulated can be transformed into a gridworld problem [33] associated with general rewards.
- We apply a pair of time difference (TD) learning algorithms for discovering the optimal policy for the problem formulated that allows the UAV to optimize its flight trajectory without the need for system identification. First, a state-action-reward-state-action (Sarsa) based UAV-trajectory optimization algorithm (SUTOA) is conceived for finding the current policy for all states and actions, which uses the Q value of the action actually chosen by the learning policy [34]. Furthermore, to enhance the learning process, we also construct a Q-learning based UAV-trajectory optimization algorithm (QUTOA), which does not require the previous Q-values.
- We analyse both the convergence and the complexity of SUTOA and QUTOA constructed for the UAV system considered. Furthermore, our simulation results demonstrate that both SUTOA and QUTOA are capable of learning the optimal trajectories for the UAV system considered. However, SUTOA and QUTOA will find different trajectories for the UAV due to their different update rules between SUTOA and QUTOA.

C. ORGANIZATION

The rest of this article is organized as follows. In Section II, the system model is presented and the problem of UAV-trajectory optimization is formulated, followed by the system analysis and the transformation for MDP are discussed in Section III. In Section IV, reinforcement learning based trajectory optimization is designed and both the convergence as well as the complexity of the learning algorithms are analyzed. Our simulation results are presented in Section V, followed by our conclusions in Section VI.

II. SYSTEM MODEL

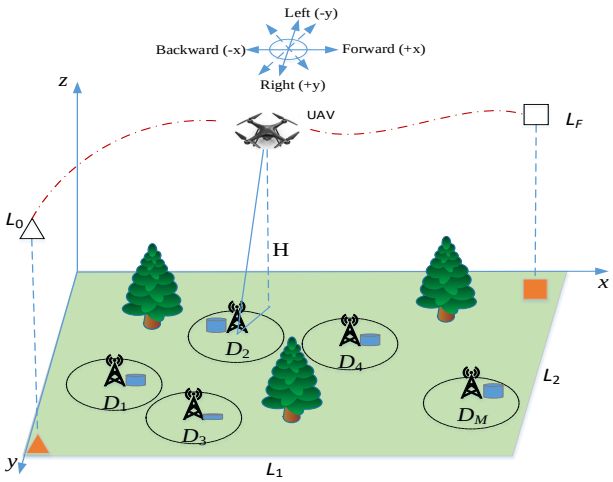


FIGURE 1: UAV enabled communication systems. The UAV flies at a fixed altitude H and the movement directions of the UAV.

We consider a UAV communication system supporting M sensors by a single UAV flying at a constant altitude of H meters (m) with speed of w, as shown in Fig. 1. We denote the sensors by $\mathcal{D} = \{D_1, D_2, \cdots, D_M\}$, which are randomly distributed across the rectangular region Ψ of L_1 by L_2 distances. For guaranteeing the communication reliability, we assume that for any sensor D_m, $D_m \in \mathcal{D}$ has a predefined communication range $\mathcal{C}_m$ based on the available radio resources. Note that the locations of the sensors may be known to the BS, but it is hard to know whether each sensor has data to transmit as well as how much data that each sensor wants to transmit. Therefore, the sensors that want to communication with the UAV as well as its data volume are unavailable to the UAV. The goal of the UAV is to collect as much data as possible by flying the region and to learn the optimal flight route within its finite flight-time T. Specifically, the UAV starts to fly from the location L_0

TABLE 1: Comparison between UAV's trajectory optimization approaches

	[22]-2008	[11]-2010	[20]-2012	[6]-2014	[16]-2016	[12]-2018	[25]-2019	[26]-2020	This article
Classical optimization		√		√	√	√			
Fully observed environment		√		√	√	√			
Partial observed environment								√	
Unknown environment	√		√				√		√
Model-based learning								√	
Model-free learning	√		√						√
Value function based learning	√		√						
Q-value based learning							√		√
Neural network based learning							√	√	

and stops at the end point L_F for each flight. Assume that the UAV applies time division multiple access (TDMA) for sequentially collecting the uploading data from the sensors. Explicitly, for each time instant t, the UAV only receives the uplink data from a single sensor. Furthermore, there are B_m bits data stored at D_m, $D_m \in \mathcal{D}$ and required to upload to the UAV.

A. SIGNAL MODEL

The locations of system can be modelled by a three-dimensional (3D) Cartesian coordinate system, where the location of D_m can be denoted as $(a_m, b_m, 0)$. The start location and the end location can be represented as $L_0 = (x_0, y_0, H)$ and $L_F = (x_F, y_F, H)$, respectively. Furthermore, the location of the UAV at time t is denoted as $(x(t), y(t), H)$. In addition, the projections of these locations to the ground plane are written as $\mathbf{g}_m = (a_m, b_m)$, $\mathbf{u}_0 = (x_0, y_0)$, $\mathbf{u} = (x_F, y_F)$ and $\mathbf{u}(t) = [x(t), y(t)]$, respectively. During the UAV's flight, the instantaneous distance between the UAV and the sensor D_m can be expressed as

$$d_m(t) = \sqrt{\|\mathbf{u}(t) - \mathbf{g}_m\|^2 + H^2}. \tag{1}$$

We assume that the UAV and all sensors have a single antenna and the communication between the UAV and the sensors are dominated by the LoS link [16]. The channel gain between the UAV and D_m is given by

$$h_m(t) = \beta_0 d_m(t)^{-\alpha} = \frac{\beta_0}{\left(\|\mathbf{u}(t) - \mathbf{g}_m\|^2 + H^2\right)^{\frac{\alpha_{pl}}{2}}}, \tag{2}$$

where β_0 denotes the channel's power loss at $d_0 = 1$ m with d_0 being the reference distance, and $\alpha_{pl} \geq 2$ is the pathloss exponent. Note that $h_m(t)$ relies on the instantaneous location of the UAV.

B. DATA TRANSMISSIONS

Since only a single sensor can successfully build communication link with the UAV for uploading its data at each time instant t, the specific sensor having the maximum channel gain to the UAV will be scheduled which is denoted as $I(t)$. Therefore, for each $t \in [0, T]$, we have $I(t) = D_m$ if

$$D_m = \min_{D_n \in \mathcal{D}} \|\mathbf{u}(t) - \mathbf{g}_n\|. \tag{3}$$

As a result, the received rate of the UAV at t can be expressed as

$$\begin{aligned} C(t) &= \log_2\left(1 + \frac{P_m |h_m(t)|^2}{\sigma^2}\right) \\ &= \log\left(1 + \frac{\gamma_0}{[(x(t) - a_m)^2 + (y(t) - b_m)^2 + H^2]^{\alpha}_{pl}}\right), \end{aligned} \tag{4}$$

where P_m denotes the transmission power of D_m, σ^2 represents the noise power, and $\gamma_0 = \frac{P\beta_0}{\sigma^2}$ is the signal-to-noise ratio (SNR). Furthermore, if a sensor such as D_m is selected, we assume $C(t) \geq r_0$ within the data collection duration in order to ensure the reception quality of the data, where r_0 is the predefined minimum target rate. Therefore, the received data rate at the UAV obeys

$$R(t) = \begin{cases} C(t), \text{ if } D_m \text{ satisfies (3) and } C(t) \geq r_0, \\ 0, \text{ otherwise.} \end{cases} \tag{5}$$

C. PROBLEM FORMULATION

This article considers the cumulative volume maximization problem of collected data in a long term by designing the trajectory of the UAV, which can be formulated as:

$$\max_{\{x(t), y(t)\}} \quad \int_0^T R(t)\mathrm{d}t \tag{6a}$$

$$\text{s.t.} \quad (x(t), y(t)) \in \Psi, t \in (0, T], \tag{6b}$$

$$\int_{t \in \mathcal{T}_m} R(t)\mathrm{d}t \leq B_m, m \in \mathcal{M}, \tag{6c}$$

$$\sqrt{(x_F - x(0))^2 + (y_F - y(0))^2} \leq wT, \tag{6d}$$

where $\mathcal{T}_m$ denotes the time duration that the UAV have to spend to collect data from D_m and w denotes the speed of the UAV; (6b) indicates that the UAV has to move within the predefined region; (6c) is to constrain the data volume that D_m can upload to the UAV. Furthermore, (6d) guarantees that the problem (6) formulated is feasible. Explicitly, given a pair

of T and w, we should guarantee that the UAV is at least able to fly from the start to the end point along the shortest path L_{min}.

Remark 1. *The constraints in* (6c) *can be treated as an incentive mechanism for reinforcement learning of the UAV, which stimulate the UAV to collect data from as many sensors as possible within T. If $B_m \to \infty$, the optimum option of the UAV is to fly over the sensor that is closest to the minimum distance of L_{min}.*

Note that when we remove the constraints in (6c), problem (6) becomes

$$\max_{\{x(t),y(t)\}} \quad \int_0^T R(t)\mathrm{d}t \tag{7a}$$

$$\text{s.t.} \quad (x(t), y(t)) \in \Psi, t \in (0, T], \tag{7b}$$

$$\sqrt{(x_F - x(0))^2 + (y_F - y(0))^2} \leq wT, \tag{7c}$$

which is equivalent to instructing the UAV to find a sensor that has the shortest distance, bearing in mind the starting point L_0 and the destination L_F. In this case, most of the sensors cannot transmit their data to the UAV.

III. SYSTEM ANALYSIS AND PROBLEM REFORMULATION

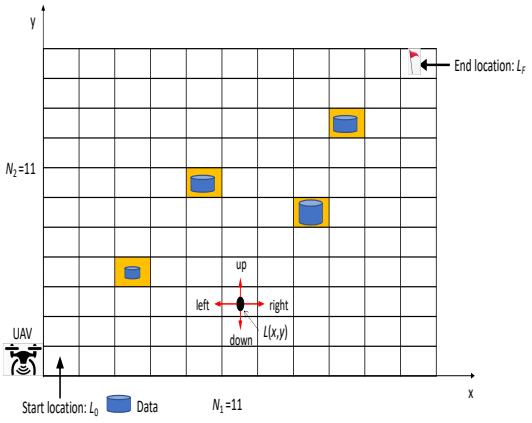
In this section, the problem (6) is transformed into a multi-period decision problem associated with finite states and actions, where the region considered is divided into perfectly tessellated tiles. Based on the transformations, the flight-time-limited UAV's trajectory optimization is modelled as a MDP.

A. REINFORCEMENT LEARNING FORMULATION

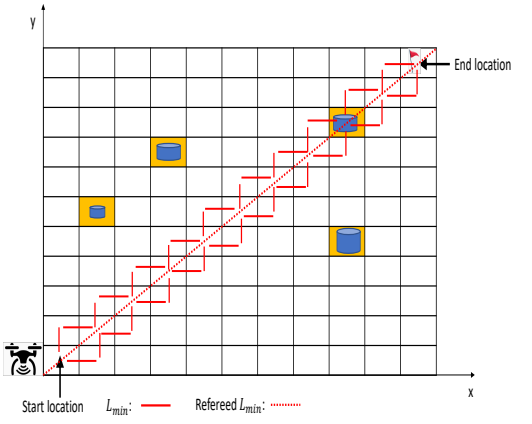
1) Agent and State Set

For solving the devised problem in (6), the UAV is considered as the learning agent who aims to learn the optimal trajectory through reinforcement learning. To model the state set of the learning agent, we first discretize the flight-time T into N time slots with step size $\tau = \frac{T}{N}$. Therefore, the region Ψ can be partitioned into $N_1 = \frac{L_1}{\tau w}$ by $N_2 = \frac{L_2}{\tau w}$ small tiles with the length of each side being τw m. Furthermore, we assume that the location of one sensor belongs to a single tile. An example of the projection on the horizontal ground plane is shown in Fig. 2, in which the system includes 4 sensors and Ψ is partitioned into 11×11 small tiles.

With the discrete tabular form for the considered region, we can represent the state set $\mathcal{S}$ of the UAV as $\mathcal{S} = \{s(1), s(2), \cdots, s(N_1 N_2)\}$ with each state $s(i)$, $s(i) \in \mathcal{S}$, referring to a small tile. We can see that there are 121 states in Fig. 2(a). Furthermore, from **Remark 1** associated with $B_m \to \infty$, we can readily observe that the optimal trajectory is represented by the red line in Fig. 2(b). However, this is not trivial for the UAV, since it cannot perceive the situation in the same way like humans. This motivates us to apply reinforcement learning for helping the UAV to make accurate decisions like humans



(a) Illustration of a fixed rate data collection and four possible directions of the UAV.



(b) The optimal trajectory with $B_m \to \infty$ for a fixed rate data collection.

FIGURE 2: A table exemplary for trajectory design [32]. Here the cylinder denotes the positions the sensors and the size of the cylinder denotes the data size.

2) Action Set

Observe from Fig. 1 and Fig. 2 that the UAV has a maximum of four actions at each state, i.e., {up, down, left, right}. For ease of clarification, we use $\mathcal{A} = \{+x, +y, -x, -y\}$ to denote the action set of the UAV, where $+x$ and $-x$ indicates that the UAV flies along the direction of left and right, respectively. Analogously, $+y$ and $-y$ indicates that the UAV flies along the direction of up and down, respectively.

Remark 2. *In practice, the UAV is capable of selecting its direction of movement along any direction θ, i.e., $\theta \in (0, 2\pi]$, which results in an infinite act of movement directions for the UAV in the action space. Based on our formulation, the*

continuous region Ψ is partitioned into N_1N_2 discrete small tiles with size τw. At a fixed UAV speed of w, the optimal trajectory can be approximated when we have $N \to \infty$.

3) Reward Formulation

The rewards of the UAV are devised to promote the quest for finding beneficial solutions that satisfy the constraints imposed on the agent in reinforcement learning [20]. In our application, the UAV aims to gather the maximum data volume from the sensors within the flight duration. In the reinforcement learning process, when the UAV takes an action a at time t and get a reward at future time t', the UAV assign the action a score for the reward according to an estimate of how important the action was in producing the reward. Moreover, we consider the fly-hover-and-communicate design as in [35], where the total data stored at a sensor will be fetched once it communicates with the UAV. More explicitly, the communication link is constructed between the UAV and a sensor, when the UAV hovers over the sensor until the data transmission is completed.

With guaranteeing the communication quality, the UAV fetches the data from the sensor only when the UAV moves to the predefine cell $cell(D_m)$, $m \in \mathcal{M}$, as shown in Fig. 2. Assume that the the UAV only receives a reward from $cell(D_m)$ so that the size of $\mathcal{C}_m$, $m \in \mathcal{M}$, will not affect the reward and can be ignored. Suppose that the UAV builds communication with D_m for data collection at time t, the received rate at the UAV can be expressed as

$$C(t) = \log\left(1 + \frac{\gamma_0}{[H^2]_{pl}^{\alpha}}\right). \tag{8}$$

Combining (5) and (8), the data volume received by the UAV at time t can be expressed as

$$R(t) = \begin{cases} C(t), & \text{if} C(t) \geq r_0 \text{ and } \sum_{i=0}^{t} C(t) \leq B_m, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Therefore, the total time taken in one tile can be expressed as

$$T_{req} = \begin{cases} \max\{\frac{B_m}{C(t)}, T_s\}, & \text{if } R(t) \neq 0, \\ T_s, & \text{otherwise,} \end{cases} \tag{10}$$

where T_s denotes the predefined time elapsed if the UAV moves from one tile to another which depends on the size of the tile as well as the speed of the UAV. Note that if the UAV fetches data from a sensor located at a tile, the time elapsed is equal to the maximum between the time required for the data transmission and the time elapsed.

Additionally, to make the UAV recognize and move to the destination during its learning, the UAV has to receive a reward $R(L_F)$ as its revenue when the UAV arrives at the destination. In this article, with maximizing the data volume collected from the sensors, we set some virtual data at the destination as the revenue to the UAV. Correspondingly, the reward $R(L_F)$ gleaned upon reaching the destination can be formulated as a function of the data volume of the sensors, i.e. $R(L_F) = f(B_1, \cdots, B_M)$. For the sake of simplicity, we consider $R(L_F) = min_{m\in\mathcal{M}} B_m$ in this article. The reward adopted and the actual data volume collected will be quantified by our simulations.

Based on the above discussions, we transform the trajectory optimization problem of the UAV into an episodic task associated with a gridworld form [33], in which the UAV starts in a start state (i.e. the start point L_0) and simulates until the terminal state (i.e. the predefined destination L_F). With limited flight-time constraint of the UAV and the characteristics of reinforcement learning, the expected discounted rewards at time t can be formulated as:

$$U(t) = \sum_{l=0}^{T} \gamma^l R(t+l+1). \tag{11}$$

Therefore, the learning form of problem (6) can be reformulated as follows:

$$\max_{\{a(t)\}} \quad U(t) \tag{12a}$$

$$\text{s.t.} \quad \sum_{t\in\mathcal{T}_m} R(t)\mathrm{d}t \leq B_m, \tag{12b}$$

$$\text{(6b) \& (6d)}, \tag{12c}$$

$$a(t) : a(t) \in \mathcal{A}, \tag{12d}$$

where $a(t)$ represents the action of the UAV taken at time t and $R(t)$ is given in (8). Note that the problem in (12) is a general learning form for UAV trajectory design. Having the dependencies amongst the associated periods, problem (12) is a multi-period decision making problem. The UAV as the decision-maker can learn the solutions of the problem via its interactions with the environment in the long run.

B. PRELIMINARIES OF MDPS

MDPs provide a mathematical framework for formulating sequential decision making problems [33], which involves delayed rewards and needs a tradeoff between immediate and delayed rewards. In this article, there are a finite number of elements in the states ($\mathcal{S}$), actions ($\mathcal{A}$) and rewards ($\mathcal{R}$), which results in a finite MDP. Hence, for specific values of $s' \in \mathcal{S}$ and $r \in \mathcal{R}$, the probability of the values occurring at a specific time t, given the values of the preceding state and action, can be expressed as

$$\begin{aligned} &P(s', r|s, a) \\ &= \Pr\{S(t) = s', R(t) = r|S(t-1) = s, A(t-1) = a\}, \end{aligned} \tag{13}$$

for all $s' \in \mathcal{S}$, $r \in \mathcal{R}$ and $a \in \mathcal{A}$. The function P describes the dynamics of the MDP and the notation $|$ represents the conditional probability. Note that the dynamics function P: $\mathcal{S} \times \mathcal{R} \times \mathcal{S} \times \mathcal{R} \to [0, 1]$ includes four arguments. Hence, P provides a probability distribution for each choice of s and a, which satisfies that

$$\sum_{s'\in\mathcal{S}} \sum_{r\in\mathcal{R}} P(s', r|s, a) = 1, \tag{14}$$

for all $s \in \mathcal{S}$ and $r \in \mathcal{R}$.

From (13), we can express the state-transition probabilities for the UAV from s to s' by taking action a, which is given by

$$\begin{aligned} P(s,a,s') &= P(s'|s,a) \\ &= \Pr\{S(t)=s'|S(t-1)=s, A(t-1)=a\} \\ &= \sum_{r\in\mathcal{R}} P(s',r|s,a), \end{aligned} \tag{15}$$

where the expected rewards for the state-action pair can be expressed as a two-argument function $r : \mathcal{S}\times\mathcal{A}\in\mathbf{R}$, i.e.,

$$\begin{aligned} R(s,a) &= \mathbb{E}[R(t)|S(t-1)=s, A(t-1)=a] \\ &= \sum_{r\in\mathcal{R}} r \sum_{s'\in\mathcal{S}} P(s',r|s,a). \end{aligned} \tag{16}$$

By using the MDP framework, reinforcement learning techniques can be exploited to solve the problem formulated by means of learning approximate solutions within a reasonable time [33]. As a benefit, the dynamics of the process do not have to be available to the UAV before taking actions. In contrast, the UAV learns both the process and the optimal policy, while interacting directly with its environment.

IV. REINFORCEMENT LEARNING BASED TRAJECTORY DESIGN

In this section, we first present the transformations of the trajectory learning algorithms adopted. Then a pair of model-free learning algorithms – SUTOA and QUTOA will be proposed for maximizing rewards received by the UAV.

A. MODEL-FREE TRAJECTORY LEARNING TRANSFORMATIONS

Based on the MDP framework, we solve the multi-period decision making problem formulated by exploiting a model-free based TD method[1], which aims to approximate the action state value function (also Q-function) via reinforcement learning. There are two classes of TD learning methods [33]: on-policy and off-policy. In the first class, the policy used for controlling the MDP is the same as the one that is being improved and evaluated. In the latter, the policy used for control, also termed as the behavior policy, can have no correlation with the policy that is being evaluated and improved, namely the estimation policy.

In reinforcement learning, a policy is defined as a mapping function from the state set to the specific probabilities of selecting each possible action[2]. Let π be the policy for the UAV, which is a probability distribution over $a\in\mathcal{A}(s)$ for each $s\in\mathcal{S}$. Explicitly, if the UAV is following policy π at time t, then $\pi(a|s)$ denotes the probability that $A(t)=a$ and $S(t)=s$. The process of reinforcement learning indicates how the UAV's policy is changed as a result of its experience. The action value function of taking action a in state s following policy π can be expressed as

$$Q_\pi(s,a) = \mathbb{E}_\pi\Big[\sum_{l=0}^{T} U(t)|S(t)=s, A(t)=a\Big], \tag{17a}$$

$$= \mathbb{E}_\pi\Big[\sum_{l=0}^{T} \gamma^l R(t+l+1)|S(t)=s, A(t)=a\Big], \tag{17b}$$

where the first equation in (17a) is the Q-function and the second equation in (17b) is the Bellman equation of the Q-function. In model-free TD learning, an optimal policy is a probability distribution with maximum Q values, when the agent starts in an arbitrary state and follows the policy thereafter. In the reinforcement learning process, the task of the UAV is to find the optimal policy that achieves the maximum reward over the long term. Specially, the relationship of a pair of policies π and π' can be described as follows [33]:

Definition 1. *Policy π is better than or as good as policy π', if the expected reward of opting for π is higher than or equal to that of pursuing π' for all states. This can be described as $\pi\geq\pi'$, if and only if $V_\pi \geq V_{\pi'}$ for all $s\in\mathcal{S}$.*

Hence, the optimal policy can be expressed as

$$\pi(s) = \arg\max_{a\in\mathcal{A}(s)} Q(s,a), \tag{18}$$

in which $\mathcal{A}(s)$ is the set of the legitimate actions at state s. Note that there may be more than one optimal policy based on the features of the problem considered, but they share the same state-value function and Q-function, denoted as Q^*_π, which can be expressed as

$$Q^*_\pi = \max_\pi Q_\pi(s,a), \text{ for all } s\in\mathcal{S} \text{ and } a\in\mathcal{A}(s). \tag{19}$$

From (17) and (19), we can obtain the Bellman optimality equation [37] for Q^*_π as follows.

$$\begin{aligned} &Q^*(s,a) \\ &= \mathbb{E}\Big[R(t+1) + \gamma \max_{a'} Q^*(S(t+1),a')|S(t)=s, A(t)=a\Big] \\ &= \sum_{s'\in\mathcal{S}} P(s'|s,a)\Big[R(s,a,s') + \gamma \max_{a'} Q^*(s',a')\Big]. \end{aligned} \tag{20}$$

To elaborate further, Fig. 3 illustrates the directed graphical representation of the reinforcement learning process of the UAV, where X and Y denote the actual environment and the environment observed by the UAV, respectively. Furthermore, the solid line represents the relationship of the operating unit in reinforcement learning, while the dashed line represents the relationships related to the actual environment. The circles around the dotted curved arrow represent a complete computation period, when the UAV takes an action, which is the minimum component of a reinforcement learning process. In Fig. 3, the UAV at time t associated with state $s(t)$ takes the action $a(t)$ and gets the reward $R(t+1)$.

[1]The TD method extracts information from observations of sequential stochastic processes in order to improve the estimates of future reactions, which is an efficient stochastic approximation technique based on samples extracted from the stochastic process that models the environment of the agent [33], [36].

[2]Note that a policy defines the learning agent's behaviour at a fixed time instant [33], which constitutes a set of rules for the agent.

Note that when $X(t) = Y(t)$, the environment is fully observed by the UAV. In this case, we can find the optimal policy by solving a set of equations, where any method of solving nonlinear equations can be used. More specifically, if there are $|\mathcal{S}|$ states and $|\mathcal{A}|$ actions for each state, then there are $|\mathcal{S}|^{|\mathcal{A}|}$ equations in the form of (20) in $|\mathcal{S}|^{|\mathcal{A}|}$ variables, i.e. (s, a) for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$. In the article, the UAV have no information about the environment, i.e., $X(t) \neq Y(t)$. Note that the information concerning Q-function are unavailable in the system considered, two approximations to (17) can be learned iteratively as follows.

1) On-Policy TD Learning

Sarsa as an on-policy method of learning the optimal policy learns the Q-value for the current policy and for all states and actions [34], [38]. The update rule for the Q-function can be expressed as

$$Q(s,a) = Q(s,a) + \alpha \left[R(s,a,s') + \gamma Q(s',a') - Q(s,a) \right]. \quad (21)$$

Note that the update equation in (21) relies on information about the variables s, s', a, a'. SUTOA continually estimates Q_π for the behaviour policy π, and at the same time changes π in a greedy manner based on Q_π. The SUTOA is concluded in **Algorithm 2**.

2) Off-Policy TD Learning

In this subsection, QUTOA is proposed, which is an off-policy learning approach for finding the optimal Q-values as well as the optimal policy of the UAV. The update rule for QUTOA is given by

$$Q(s,a) = Q(s,a) + \alpha \left[R(s,a,s') + \gamma \max_{a' \in \mathcal{A}(s')} Q(s',a') - Q(s,a) \right]. \quad (22)$$

The Q-learning based UAV-trajectory optimization process is concluded in **Algorithm 1**. In fact, the actions chosen can be based on some other policy that may have no relationship with $Q_{ql}(s)$. For instance, a policy associated with a uniform distribution over the action space can be used for generating the actions. However, the random behavior policy usually generates a poor control performance for the MDP, as we will demonstrate in Section V.

Algorithm 1: Sarsa based UAV-trajectory learning process

input : Parameters for learning: $\gamma \in [0,1]$, $\alpha \in (0,1]$, $\epsilon \in [0,1]$); Agent information: t_s, L_0, L_F, T, N_1, N_2, w, H. $ep = 0$
output: Rewards: R_{sarsa}; Optimal policies: π^*_{sarsa}.

```
while ep ≤ maximum episodes do
    Initialize Q_sarsa(s,a) = 0, for all s and a;
      s = s' = L_0, T_s = 0, ep = 0, T_p = 0;
    For state s, the UAV chooses action a according to
      the policy derived from Q_sarsa(s) of (23)
    repeat
        The UAV takes action a ;
        The UAV receives a reward R, moves to a
          successor state s' and observes the elapsed
          T_s;
        Choose an allowed action a' from the state s'
          following the policy derived from Q_sarsa(s')
          as in (23) ;
        Update the Q-values based on (21), we have
          Q_sarsa(s,a) ←
          Q_sarsa(s,a) + α[R(s,a,s') +
          γQ_sarsa(s',a') − Q_sarsa(s,a)];
        Update the state, the action and the elapsed
          time: s ← s', a = a'; T_p ← T_p + T_req;
    until s' == L_F or T_p >= T;
    ep ← ep + 1;
end
```

B. ACTION SELECTION POLICIES

An important component of both on-policy and off-policy learning algorithms is the action selection policy, which is used for generating a sequence of actions that the UAV will perform during the learning process. Its objective is to ensure the success of reinforcement learning by striking a tradeoff between exploration and exploitation during the learning process [33], where the exploration allows the learning process getting trapping out of a local optimum, while the exploitation prompts the convergence of the learning process. More specifically, the exploration motivates the learning agent (the UAV) to learn from the environment, in which the UAV opts for a specific action, observes a certain reward and then updates its action choices. On the other hand, the exploitation helps the UAV take advantage of the knowledge that is already available to make the best action choice. Therefore, our goal is to choose an action selection method that allows the UAV to reinforce beneficial actions, whilst also exploring new actions during the learning process. In this article, we consider the so-called ϵ-greedy exploration [39], in which the probability of choosing action a at state s can be expressed as

$$a^* = \begin{cases} \arg\max_{a \in \mathcal{A}(s)} Q(s,a), & \text{with probability } 1-\epsilon, \\ \text{random selection}, & \text{with probability } \epsilon. \end{cases} \quad (23)$$

C. ANALYSIS OF THE PROPOSED ALGORITHMS

In this subsection, we investigate the convergence and the complexity of the proposed reinforcement learning algorithms for the UAV system considered.

1) Convergence Analysis

Note that in **Algorithm 1** and **Algorithm 2**, the learning processes both in SUTOA and QUTOA visit all possible Q-values, which allows convergence to an optimal policy, avoiding getting stuck in sub-optimal policies. Moreover, for

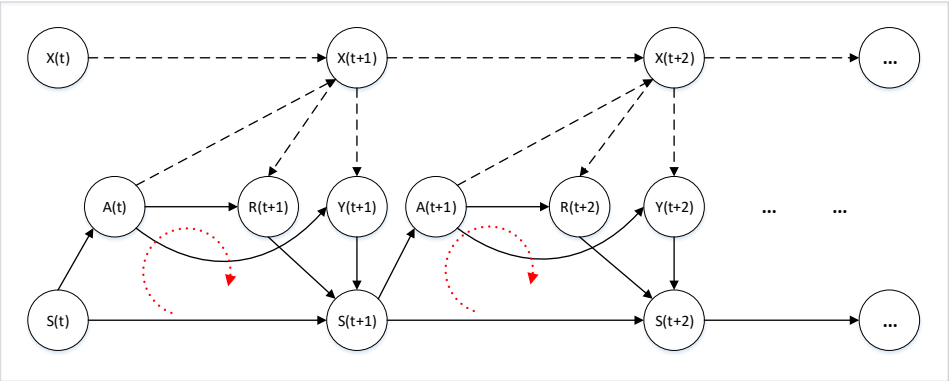


FIGURE 3: Directed graphical representation of the UAV trajectory optimization considered.

Algorithm 2: Q-learning based trajectory learning process

input : Parameters for learning: $\gamma \in [0, 1]$, $\alpha \in (0, 1]$, $\epsilon \in [0, 1]$); Agent information: t_s, L_0, L_F, T, N_1, N_2, w, H. $ep = 0$

output: Rewards: R_{ql}; Optimal policies: π^*_{ql}.

while *ep* $\leq$ *maximum episodes* **do**
  Initialize $Q_{ql}(s, a) = 0$, for all s and a;
  $s = s' = L_0$, $T_s = 0$, $ep = 0$, $T_p = 0$;
  repeat
    For state s, the UAV chooses action a according to the policy derived from $Q(s)$ of (23) ;
    The UAV takes an action a, receives a reward R, moves to a next state s' and observes the elapsed time T_s;
    Update Q-values based on (22), we have $Q_{ql}(s, a) \leftarrow Q_{ql}(s, a) + \alpha\Big[R(s, a, s') + \gamma \max_{a' \in \mathcal{A}(s')} Q_{ql}(s', a') - Q_{ql}(s, a)\Big]$;
    Update the state, the action and the elapsed time: $s \leftarrow s'$, $a = a'$; $T_p \leftarrow T_p + T_{req}$;
  until $s' == L_F$ *or* $T_p >= T$;
  $ep \leftarrow ep + 1$;
end
Find the optimal policy:
$\pi^*_{sarsa}(s) = \arg\max_a Q_{sarsa}(s, a)$ for any $s \in \mathcal{S}$.
$\pi^*_{ql}(s) = \arg\max_a Q_{ql}(s, a)$ for any $s \in \mathcal{S}$.

each episode both SUTOA and QUTOA in **Algorithm 1** and **Algorithm 2**, will stop when the UAV reaches its final destination point or exhausts its maximum flying time. If the algorithm stops by satisfying the first condition, i.e., the UAV reaches its destination point, this indicates that the UAV has learned a path by using the proposed reinforcement learning algorithms. The second stopping condition of $T_p > T$ is used for ensuring that the learning process cannot exceed the maximum flying time of the UAV in each episode. In order to guarantee a high learning performance, the flight-time is assumed to be higher than the learning time of a single episode. Therefore, we focus on the proof of convergence for SUTOA and QUTOA in **Algorithm 1** and **Algorithm 2**, based on the first stopping condition. Although both SUTOA and QUTOA estimate the optimal Q-value by iterating, SUTOA is an on-policy algorithm, hence its convergence relies on the learning policy found. The convergence behaviour of QUTOA for solving the UAV system considered that satisfies finite MDPs can be found in [19], [40]. In this article, an ϵ-greedy learning policy is considered, which satisfies the condition of greedy in the limit with infinite exploration (GLIE) [41]. Hence the convergence of SUTOA can be characterized by the following remark.

Remark 3. *Let us assume that the UAV's trajectory optimization problem is modelled by a finite MDP and fix a greedy policy as in* ***Algorithm 1****, of Section IV-B. SUTOA converges to the optimal Q-value Q^* and the learning policy converges to an optimal policy π^*, if the following conditions are satisfied:*

1) *The Q-values of SUTOA are stored in a lookup table, which means that no function approximations are used;*
2) *The learning rate of SUTOA satisfies* $0 \leq \alpha \leq 1$ $\sum_{t=0}^{+\infty} \alpha = \infty$ *and* $\sum_{t=0}^{+\infty} \alpha^2 < \infty$*;*
3) $\mathrm{Var}[R(s, a)]$ *is bounded for all state-action pairs* $(s, a) \in \mathcal{S} \times \mathcal{A}$*, where* Var *denotes the variance of the reward* $R(s, a)$.

2) Complexity Analysis

In this article, we portray the task of the UAV as the gridworld of finding the optimal policy of the UAV. In **Algorithm 1** and **Algorithm 2**, the trajectory learning processes both in SUTOA and QUTOA visit all possible Q-values based on their update equations within a single episode. Let $n = |\mathcal{S}|$ and $e = \sum_{s \in \mathcal{S}} |\mathcal{A}(s)|$ represent the total number of states for the UAV and the total number of actions for the UAV, respectively. Note that in the formulated reinforcement learning problem for the UAV considered, we assume that the action space is deterministic, i.e., all actions are known to the UAV and the state space is observable for the UAV. This indicates

TABLE 2: Simulation parameters

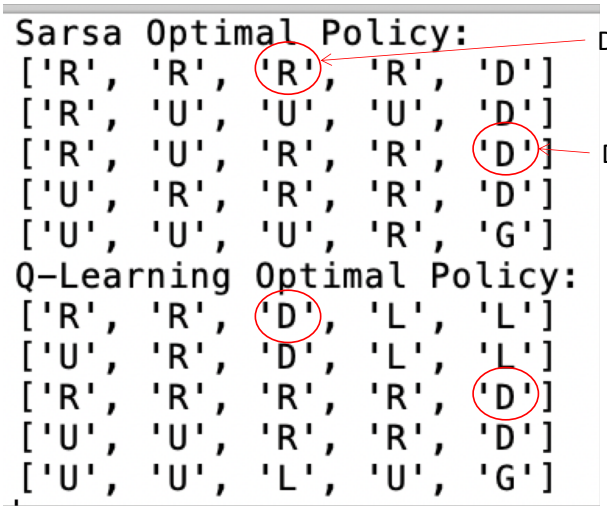
Parameter	Value
Area of the considered region	10×10 m^2
Number of sensors M	2,4,5
Maximum flight time of the UAV T	30 mins
Altitude of UAV H	100 m
Speed of the UAV w	20 m/s
Noise power σ^2	-110 dBm
Channel power gain β_0	-50 dB
Path loss exponent α_{pl}	2
The transmit power P_m	10 dBm
Bandwidth B	2 MHz
(T_s, N_1)	(0.5s, 5) and (0.1s, 11)
Learning rate α	0.5
Discounting factor γ	0.98

that the UAV is capable of perfectly determining its current state, including whether it is currently achieving its goal. In this case, the number of iterations of QUTOA is at most on the order of $\mathcal{O}(e \cdot n)$ steps [42]. In particular, the worst-case complexity of QUTOA can be expressed as $\mathcal{O}(n^3)$, if a state space has no duplicate actions, i.e., $e \leq n^2$. More specifically, in the problem formulated in Section III, the UAV has at most four actions in each sate, hence we have $e \leq 4n$ and the worst-case complexity becomes $\mathcal{O}(4n^2)$. This means that the complexity of QUTOA is polynomial in n, even if it uses undirected exploration during its iterations. Moreover, SUTOA has the same computational complexity, $\mathcal{O}(en)$ as QUTOA [43], but the update in SUTOA is based on on-policy samples, hence it may impose a high sample complexity. Finally, it should be pointed out that the number of states is an exponential function of the number of state variables. Hence, the computational complexity of both SUTOA and QUTOA will increase rapidly, when the state space becomes large.

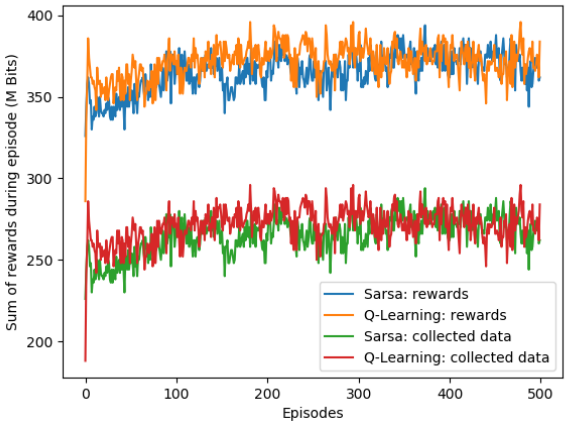
V. SIMULATION RESULTS

In this section, we characterize the performance of our UAV system using the two different learning approaches. The simulation parameters are given in Table. 2. We first consider the scenario of $N_1 = N_2 = 5$ and $|\mathcal{S}| = 25$ states in total, where the size of the region considered is $10 \times 10\text{m}^2$. There are two sensors, namely D_1 has $B_1 = 100$ Mbits of data and D_2 has $B_2 = 200$ Mbits of data, which are randomly located in the region considered.

In particular, a specific scenario with two sensors D_1 and D_2 is illustrated in Fig. 4, where the optimal trajectories are found using SUTOA and QUTOA, separately. Note that the differences of the update rues on Sarsa and Q-learning lead to Sarsa is more conservative than Q-learning when it explored the actions, which resulted in Sarsa and Q-learning will find different solution [33]. Fig. 4(a) shows the optimal policies for the two different algorithms. Fig. 4(b) shows the cumulative rewards and the true amount of the data collected at the UAV using different algorithms. From Fig. 4(b), we can see that in the two learning algorithms, the true amount of the data collected by the UAV has the same increasing trend as the sum of rewards, which indicates that the reward function



(a) Optimal policy for different algorithms, where U, D, L, R denote up, down, left, right, respectively.



(b) Cumulative Rewards received by the UAV using different algorithms in different episodes

FIGURE 4: Learning results of different algorithms in a 5×5 grid region, where $\gamma = 0.98$ and $\alpha = 0.5$.

is capable of resulting in the good learning performance. Furthermore, since a virtual reward will be received by the UAV when it arrives at the destination, the sum of rewards is higher than the true amount of the collected data.

In Fig. 5, we investigate the sum rewards of reinforcement learning based on SUTOA and QUTOA proposed in **Algorithm 1** and **Algorithm 2**, where these data were collected over 50 runs and using $\gamma = 1$ corresponding to a non-discounted scenario. For comparisons, we also consider an off-policy scheme for QUTOA, in which the actions are generated by the policy that follows the uniform distribution on the action space. As observed from Fig. 5, the sum rewards attained by SUTOA is higher than that attained by QUTOA. This is because SUTOA takes the action selection into account and learns a near-optimal policy whilst exploring,

which can be found from Step 6 - Step 7 of **Algorithm 1**. Moreover, QUTOA with random action selection has lower sum rewards than the other two algorithms.

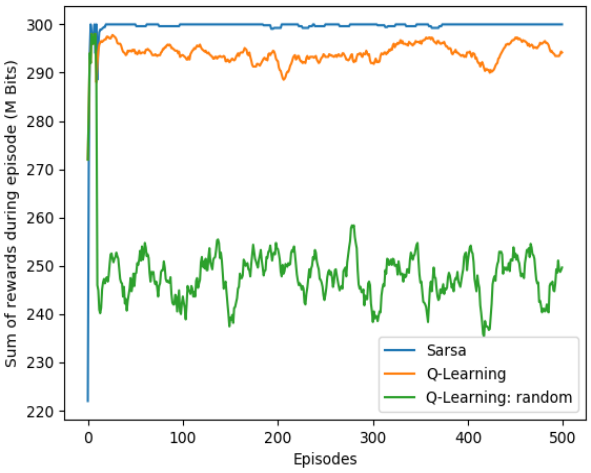


FIGURE 5: Sum of rewards received by different algorithms in different episodes.

Fig. 6 illustrates the sum of rewards received by different algorithms during the episodes associated with $N_1 = N_2 = 11$ and two sensors randomly distributed across the tiles. As seen from Fig. 6, QUTOA has a significant advantage over SUTOA in this case due to the limited flight-time T of the UAV. Specifically, during the process that the UAV learns the optimal policy, SUTOA requires more actions than QUTOA, which can be seen from Step 7 for SUTOA and Step 5 for QUTOA in **Algorithm 2**. As a result, the UAV cannot reach the destination in SUTOA when the flight-time is not sufficiently long. That is the loop in SUTOA would terminate by the second condition in Step 9 of **Algorithm 1**. Moreover, from Fig. 6, we can observe that QUTOA is a better option than SUTOA for learning the UAV's trajectory in the system considered. This is because the reward received by the UAV in this article is a positive feedback from the environment and the punishments for taking wrong actions by the UAV are not included [33].

In Fig. 7, we investigate the average sum rewards of reinforcement learning based on SUTOA and QUTOA proposed in **Algorithm 1** and **Algorithm 2** with $N_1 = N_2 = 11$. Furthermore, different number of sensors are considered with $M = 4$ and $M = 5$, which are randomly distributed in the grids with each sensor having $B = 200$ Mbits of data for transmission. As observed from Fig. 7, QUTOA is a beneficial option for the data collection problem in the system considered, when finding the optimal policy under flight-time constraints.

VI. CONCLUSIONS

In this article, we invoked the reinforcement learning for the UAV's trajectory optimization with the goal of maximizing

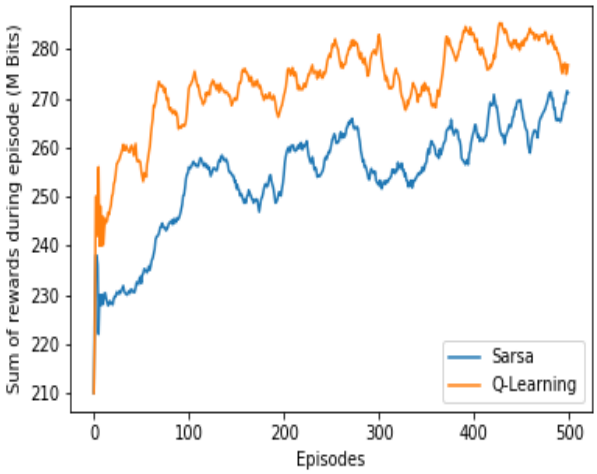
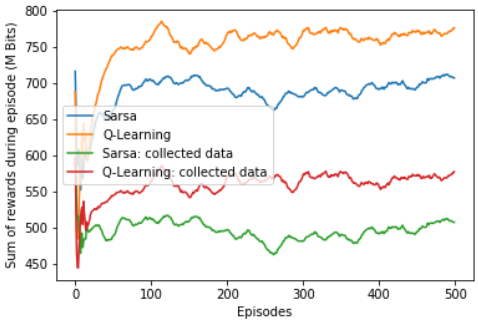
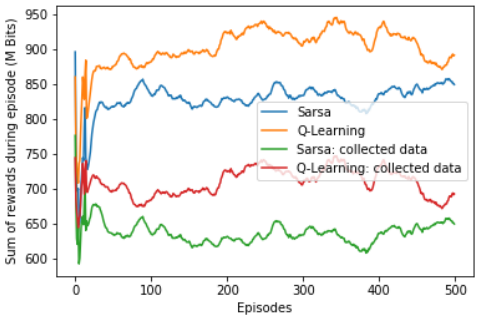


FIGURE 6: Sum of rewards received by different algorithms during episodes with $\gamma = 0.98$.



(a) $M = 4$



(b) $M = 5$

FIGURE 7: Sum of rewards and sum of collected data during episodes with different algorithms, where $N_1 = N_2 = 11$ and $B = 200$Mbits.

the cumulative data volume fetched from the sensors, where the network information are unavailable to the UAV, such as the locations of the sensors and the amount of data to be transmitted. Given the associated uncertainties and chal-

lenges, we transform the trajectory optimization of the UAV into a finite MDP by dividing the region considered into small tiles. Furthermore, model-free reinforcement learning approaches were utilized to solve the MDP problem formulated, which allowed the UAV to optimize its flight trajectory without the need for system identification. Specifically, we developed SUTOA and QUTOA for finding the optimal trajectory. Our simulation results revealed that SUTOA and QUTOA could find an optimal trajectory under our flight-time constraint with the rewards defined. Furthermore, since SUTOA is more conservative than QUTOA when it explored the actions, which resulted in SUTOA performing a little better, when some penalties were encountered between the starting point and the destination. A promising extension of this work is to consider model-based approaches such as Dyna-Q based UAV-trajectory learning based on the idea of combining experience and model. When the state space and the action space are continuous or have multiple dimensions, deep Q-learning based UAV-trajectory optimization becomes a promising technique of circumventing to handle the challenges by using a deep neural network to approximate the Q-function, which is another promising research direction.

REFERENCES

[1] T. 38.811. (2019) Study on new radio (NR) to support non-terrestrial networks (Release 15). [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3234

[2] S. Chandrasekharan, K. Gomez, A. Al-Hourani, S. Kandeepan, T. Rasheed, L. Goratti, L. Reynaud, D. Grace, I. Bucaille, T. Wirth, and S. Allsopp, "Designing and implementing future aerial communication networks," IEEE Commun. Mag., vol. 54, no. 5, pp. 26–34, May 2016.

[3] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," IEEE J. Sel. Areas Commun., vol. 35, no. 5, pp. 1046–1061, 2017.

[4] A. H. Michel, "Amazon's drone patents," Center for the Study of the Drone at Bard College, http://dronecenter. bard. edu/amazon-drone-patents, 2017.

[5] Q. Liu, J. Wu, P. Xia, S. Zhao, W. Chen, Y. Yang, and L. Hanzo, "Charging unplugged: Will distributed laser charging for mobile wireless power transfer work?" IEEE Veh. Technol. Mag., vol. 11, no. 4, pp. 36–45, Dec. 2016.

[6] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," IEEE Wireless Commun. Lett., vol. 3, no. 6, pp. 569–572, Dec. 2014.

[7] X. Lin, V. Yajnanarayana, S. D. Muruganathan, S. Gao, H. Asplund, H. Maattanen, M. Bergstrom, S. Euler, and Y. . E. Wang, "The sky is not the limit: LTE for unmanned aerial vehicles," IEEE Commun. Mag., vol. 56, no. 4, pp. 204–210, Apr. 2018.

[8] F. Cheng, S. Zhang, Z. Li, Y. Chen, N. Zhao, F. R. Yu, and V. C. M. Leung, "UAV trajectory optimization for data offloading at the edge of multiple cells," IEEE Trans. Veh. Technol., vol. 67, no. 7, pp. 6732–6736, Jul. 2018.

[9] M. M. Amiri and D. Gunduz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," CoRR, vol. abs/1901.00844, 2019. [Online]. Available: http://arxiv.org/abs/1901.00844

[10] Y. Zeng, X. Xu, and R. Zhang, "Trajectory design for completion time minimization in UAV-enabled multicasting," IEEE Trans. Wireless Commun., vol. 17, no. 4, pp. 2233–2246, Apr. 2018.

[11] R. Sugihara and R. K. Gupta, "Speed control and scheduling of data mules in sensor networks," ACM Trans. Sen. Netw., vol. 7, no. 1, pp. 4:1–4:29, Aug. 2010. [Online]. Available: http://doi.acm.org/10.1145/1806895.1806899

[12] J. Gong, T. Chang, C. Shen, and X. Chen, "Flight time minimization of UAV for data collection over wireless sensor networks," IEEE J. Sel. Areas Commun., vol. 36, no. 9, pp. 1942–1954, Sep. 2018.

[13] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," IEEE Trans. Veh. Technol., vol. 67, no. 3, pp. 2049–2063, Mar. 2018.

[14] L. Ge, P. Dong, H. Zhang, J. Wang, and X. You, "Joint beamforming and trajectory optimization for intelligent reflecting surfaces-assisted uav communications," IEEE Access, pp. 1–1, 2020.

[15] C. O. Nnamani, M. R. A. Khandaker, and M. Sellathurai, "Uav-aided jamming for secure ground communication with unknown eavesdropper location," IEEE Access, vol. 8, pp. 72 881–72 892, 2020.

[16] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for UAV-enabled mobile relaying systems," IEEE Trans. Commun., vol. 64, no. 12, pp. 4983–4996, Dec. 2016.

[17] N. Mohamed, J. Al-Jaroodi, I. Jawhar, H. Noura, and S. Mahmoud, "UAVFog: A UAV-based fog computing for internet of things," in SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI, Aug. 2017, pp. 1–8.

[18] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," IEEE Commun. S. Tut., vol. 21, no. 4, pp. 3039–3071, 2019.

[19] J. Cui, Y. Liu, and A. Nallanathan, "Multi-agent reinforcement learning based resource allocation for UAV networks," IEEE Trans. Wireless Commun., pp. 1–1, 2019.

[20] B. Pearre and T. X. Brown, "Model-free trajectory optimisation for unmanned aircraft serving as data ferries for widespread sensors," Remote Sensing, vol. 4, no. 10, pp. 2971–3005, Oct. 2012.

[21] H. Bayerlein, P. D. Kerret, and D. Gesbert, "Trajectory optimization for autonomous flying base station via reinforcement learning," in International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Jun. 2018, pp. 1–5.

[22] D. Henkel and T. X. Brown, "Towards autonomous data ferry route design through reinforcement learning," in International Symposium on a World of Wireless, Mobile and Multimedia Networks, Jun. 2008, pp. 1–6.

[23] D. WR Jr, "Application of artificial intelligence techniques in uninhabited aerial vehicle flight," in Digital Avionics Systems Conference (DASC), vol. 2. IEEE, Oct. 2003, pp. 8.C.3–1–8.C.3–6.

[24] K. Sundaresan, E. Chai, A. Chakraborty, and S. Rangarajan, "SkyLiTE: End-to-end design of low-altitude UAV networks for providing LTE connectivity," CoRR, vol. abs/1802.06042, 2018. [Online]. Available: http://arxiv.org/abs/1802.06042

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

[26] H. Shiri, J. Park, and M. Bennis, "Communication-efficient massive UAV online path control: Federated learning meets mean-field game theory," 2020.

[27] J. Hu, H. Zhang, and L. Song, "Reinforcement learning for decentralized trajectory design in cellular uav networks with sense-and-send protocol," IEEE Internet of Things Journal, vol. 6, no. 4, pp. 6177–6189, 2019.

[28] C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, "Energy-efficient uav control for effective and fair communication coverage: A deep reinforcement learning approach," IEEE Journal on Selected Areas in Communications, vol. 36, no. 9, pp. 2059–2070, 2018.

[29] J. Hu, H. Zhang, L. Song, Z. Han, and H. V. Poor, "Reinforcement learning for a cellular internet of uavs: Protocol design, trajectory control, and resource management," IEEE Wireless Commun., vol. 27, no. 1, pp. 116–123, 2020.

[30] 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.

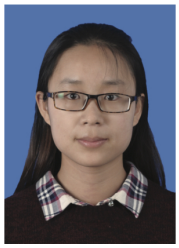
[31] M. Chen, W. Saad, and C. Yin, "Liquid state machine learning for resource and cache management in LTE-U unmanned aerial vehicle (UAV) networks," IEEE Trans. Wireless Commun., vol. 18, no. 3, pp. 1504–1517, Mar. 2019.

[32] J. Cui, Z. Ding, Y. Deng, and A. Nallanathan, "Model-free based automated trajectory optimization for UAVs toward data transmission," in IEEE Proc. of Global Commun. Conf. (GLOBECOM), Dec. 2019.

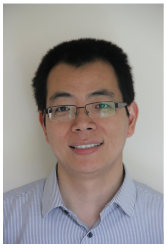
[33] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction (second edition). Cambridge, MA: MIT Press, 2018. [Online]. Available: http://incompleteideas.net/book/the-book.html

[34] S. P. Singh and R. S. Sutton, "Reinforcement learning with replacing eligibility traces," Machine Learning, vol. 22, no. 1, pp. 123–158, Mar 1996. [Online]. Available: https://doi.org/10.1007/BF00114726

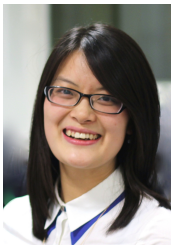
[35] H. He, S. Zhang, Y. Zeng, and R. Zhang, "Joint altitude and beamwidth optimization for uav-enabled multiuser communications," IEEE Commun. Lett., vol. 22, no. 2, pp. 344–347, Feb. 2018.
[36] P. Dayan and T. J. Sejnowski, "TD(λ) converges with probability 1," Machine Learning, vol. 14, no. 3, pp. 295–301, Mar 1994. [Online]. Available: https://doi.org/10.1007/BF00993978
[37] R. A. Howard, Dynamic programming and Markov Decision processes. MIT Press., 1960.
[38] G. Rummery and M. Niranjan, "On-line Q-learning using connectionist systems," Technical Report CUED/F-INFENG/TR 166, Nov. 1994.
[39] C. J. C. H. Watkins, "Learning from delayed rewards," Ph.D. dissertation, King's College, Cambridge, UK, May 1989. [Online]. Available: https://www.cs.rhul.ac.uk/home/chrisw/thesis.html
[40] T. Jaakkola, M. I. Jordan, and S. P. Singh, "Convergence of stochastic iterative dynamic programming algorithms," in Neural Computation, vol. 6, 1994, pp. 1185–1201.
[41] S. Singh, T. Jaakkola, M. L. Littman, and C. Szepesvári, "Convergence results for single-step on-policy reinforcement-learning algorithms," Machine Learning, vol. 38, no. 3, pp. 287–308, Mar 2000. [Online]. Available: https://doi.org/10.1023/A:1007678930559
[42] S. Koenig and R. G. Simmons, "Complexity analysis of real-time reinforcement learning," in AAAI, 1993, pp. 99–107.
[43] A. Geramifard, T. J. Walsh, T. Stefanie, G. Chowdhary, N. Roy, and J. P. How, A Tutorial on Linear Function Approximators for Dynamic Programming and Reinforcement Learning. now, 2013. [Online]. Available: https://ieeexplore.ieee.org/document/8187372



JINGJING CUI (M'18) received the Ph.D from Southwest Jiaotong University, Chengdu, China 2018. She is currently a research fellow with the School of Electronics and Computer Science, University of Southampton, UK. She was a research assistant with the School of Electronic Engineering and Computer Science, Queen Mary University of London, UK, from May 2018 to May 2019. Her research interests include optimization theory and algorithm design, machine learning for wireless networks, and quantum communications. She received the Exemplary Reviewers of the IEEE Transactions on Communications in 2019 and IEEE Communication Letters in 2018. She has also served as a TPC Member for IEEE conferences, such as ICC-2020 and IEEE GLOBECOM-2020.



ZHIGUO DING (S'03-M'05-F'20) received his Ph.D from Imperial College London in 2005. He is currently a Professor in Communications at the University of Manchester. From Sept. 2012 to Sept. 2019, he has also been an academic visitor in Princeton University. Dr Ding' research interests are 5G networks, game theory, cooperative and energy harvesting networks and statistical signal processing. He has been serving as an Editor for IEEE TCOM, IEEE TVT, and served as an editor for IEEE WCL and IEEE CL. He received the best paper award in ICWMC-2009 and WCSP-2015, IEEE Communication Letter Exemplary Reviewer 2012, the EU Marie Curie Fellowship 2012-2014, IEEE TVT Top Editor 2017, 2018 IEEE Communication Society Heinrich Hertz Award, 2018 IEEE Vehicular Technology Society Jack Neubauer Memorial Award, and 2018 IEEE Signal Processing Society Best Signal Processing Letter Award.



YANSHA DENG (S13-M20) received the Ph.D. degree in electrical engineering from the Queen Mary University of London, U.K., in 2015. From 2015 to 2017, she was a Post-Doctoral Research Fellow with Kings College London, U.K., where she is currently a Lecturer (Assistant Professor) with the Department of Informatics. Her research interests include molecular communication, machine learning, and 5G wireless networks. She was a recipient of the Best Paper Awards from ICC 2016 and Globecom 2017 as the first author. She is currently an Associate Editor of the IEEE Transactions on Communications, IEEE Transactions on Molecular, Biological and Multi-scale Communications, and the Senior Editor of the IEEE Communication Letters. She also received the Exemplary Reviewers of the IEEE Transactions on Communications in 2016 and 2017, and IEEE Transactions on Wireless Communications in 2018. She has also served as a TPC Member for many IEEE conferences, such as IEEE GLOBECOM and ICC.

ARUMUGAM NALLANATHAN (S'97-M'00-SM'05-F'17) is Professor of Wireless Communications and Head of the Communication Systems Research (CSR) group in the School of Electronic Engineering and Computer Science at Queen Mary University of London since September 2017. He was with the Department of Informatics at King's College London from December 2007 to August 2017, where he was Professor of Wireless Communications from April 2013 to August 2017 and a Visiting Professor from September 2017. He was an Assistant Professor in the Department of Electrical and Computer Engineering, National University of Singapore from August 2000 to December 2007. His research interests include Artificial Intelligence for Wireless Systems, Beyond 5G Wireless Networks, Internet of Things (IoT) and Molecular Communications. He published nearly 500 technical papers in scientific journals and international conferences. He is a co-recipient of the Best Paper Awards presented at the IEEE International Conference on Communications 2016 (ICC'2016) , IEEE Global Communications Conference 2017 (GLOBECOM'2017) and IEEE Vehicular Technology Conference 2018 (VTC'2018). He is an IEEE Distinguished Lecturer. He has been selected as a Web of Science Highly Cited Researcher in 2016.

He is an Editor for IEEE Transactions on Communications. He was an Editor for IEEE Transactions on Wireless Communications (2006-2011), IEEE Transactions on Vehicular Technology (2006-2017), IEEE Wireless Communications Letters and IEEE Signal Processing Letters. He served as the Chair for the Signal Processing and Communication Electronics Technical Committee of IEEE Communications Society and Technical Program Chair and member of Technical Program Committees in numerous IEEE conferences. He received the IEEE Communications Society SPCE outstanding service award 2012 and IEEE Communications Society RCC outstanding service award 2014.

LAJOS HANZO (M'91-SM'92-F'04) (http:www-mobile.ecs.soton.ac.uk, https:en.wikipedia.org/wiki/Lajos_Hanzo) FREng, FIEEE, FIET, Fellow of EURASIP, DSc has received his Master degree and Doctorate in 1976 and 1983, respectively from the Technical University (TU) of Budapest. He was also awarded Honorary Doctorates by the TU of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE Press. He has served as Governor of both IEEE ComSoc and of VTS. He has published 1900+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 119 PhD students. Over 40 of them are Professors at various stages of their careers in academia and many of them are leading scientists in the wireless industry.