# Reinforcement Learning for a Cellular Internet of UAVs: Protocol Design, Trajectory Control, and Resource Management

Jingzhi Hu, Hongliang Zhang, Lingyang Song, Zhu Han, and H. Vincent Poor

## Abstract

Unmanned aerial vehicles (UAVs) can be powerful Internet of Things components to execute sensing tasks over the next-generation cellular networks, which are generally referred to as the cellular Internet of UAVs. However, due to the high mobility of UAVs and shadowing in air-to-ground channels, UAVs operate in a dynamic and uncertain environment. Therefore, UAVs need to improve the quality of service of sensing and communication without complete information, which makes reinforcement learning suitable for use in the cellular Internet of UAVs. In this article, we propose a distributed sense-and-send protocol to coordinate UAVs for sensing and transmission. Then we apply reinforcement learning in the cellular Internet of UAVs to solve key problems such as trajectory control and resource management. Finally, we point out several potential future research directions.

## Introduction

The emerging unmanned aerial vehicles (UAVs) have been playing an increasing role in military, public, and civil applications [1]. Specifically, exploiting UAVs as Internet of Things (IoT) devices to execute sensing tasks has been of particular interest due to its advantages of on-demand flexible deployment, large service coverage, and ability to hover at a high altitude [2]. Such sensing tasks consist of a wide range of critical daily applications, for example, smart agriculture, security monitoring, forest fire detection, and traffic surveillance, as illustrated in Fig. 1. To realize the above vision, it is envisaged by the Third Generation Partnership Project (3GPP) that cellular networks are necessary for UAVs to execute sensing tasks, which we refer to as the *cellular Internet of UAVs* [3].

In the cellular Internet of UAVs, UAVs sense the targets of tasks and then transmit sensory data to the base stations (BSs) immediately. Therefore, the sensing and transmission tasks of UAVs are coupled [4]. Moreover, due to the high mobility of UAVs and shadowing in air-to-ground channels, UAVs operate in a dynamic and uncertain environment [5]. Therefore, UAVs must improve their quality of service (QoS) in both sensing and transmission without complete information. Due to incomplete information, the coordination of multiple UAVs to execute sensing tasks is a challenging problem.

In this article, we introduce reinforcement learning approaches and their applications in the cellular Internet of UAVs. Since reinforcement learning can enable UAVs to improve their policies to achieve objectives without a priori knowledge or complete information of the environment, it is suitable to address the key problems in the cellular Internet of UAVs [6]. We focus on the following three essential parts of the cellular Internet of UAVs:

- *Protocol Design:* We present a distributed sense-and-send protocol to coordinate the UAVs in sensing and transmission.
- *Trajectory Control:* We discuss the dynamic trajectory control problem of UAVs and propose an enhanced multi-UAV Q-learning algorithm for this problem.
- *Resource Management:* We introduce different reinforcement learning approaches and their applications for resource management problems, including user association, power management, and subchannel allocation.

Specifically, to address the trajectory control and resource management problems, we discuss the possible implementations of reinforcement learning approaches in the cellular Internet of UAVs:

- Applying multi-armed bandit learning to solve the user association problem
- Utilizing Q-learning to solve the trajectory control problem
- Using actor-critic learning to solve the power management problem
- Applying deep reinforcement learning to solve the subchannel allocation problem

The rest of the article is organized as follows. First, we provide an overview of the cellular Internet of UAVs and demonstrate the sense-and-send protocol. Then we discuss the reinforcement learning approaches, including the basics and applications in the cellular Internet of UAVs. Following that, we elaborate on how to apply Q-learning to solve the UAV trajectory control problem. Finally, we draw conclusions and point out several future research directions.

Digital Object Identifier:
10.1109/MWC.001.1900262

*Jingzhi Hu and Lingyang Song are with Peking University; Hongliang Zhang is with Peking University, and also with the University of Houston; Zhu Han is with the University of Houston, and also with Kyung Hee University; H. Vincent Poor is with Princeton University.*


1536-1284/20/$25.00 © 2020 IEEE

## Overview of the Cellular Internet of UAVs

In this section, we first introduce the cellular Internet of UAVs. Then, to coordinate multiple UAVs to execute sensing tasks, we propose a distributed sense-and-send protocol.

### Cellular Internet of UAVs

As shown in Fig. 2, in a cellular Internet of UAVs, multiple UAVs execute a set of sensing tasks. Each task has one target to be sensed, and the targets are at different locations. The tasks are pre-assigned to the UAVs, and the UAVs sense the targets of their tasks and transmit the results to the BSs continuously. To be specific, the UAVs execute the tasks through two steps: *UAV sensing* and *UAV transmission*.

**UAV Sensing:** Each UAV is equipped with an onboard sensor to sense its target. Due to the limited sensing capability of the sensor, the sensing is not always successful.[1] If the sensing is successful, the sensory data collected by the UAV is referred to as valid; otherwise, it is referred to as invalid. In general, the probability of successful sensing is negatively related to the distance between the sensor and the target [7].

**UAV Transmission:** Each UAV is associated with one BS and uses the uplink subchannels allocated by the BS to transmit the sensory data.[2] Each BS owns a limited number of subchannels to support UAV transmission. The frequency bands used by different BSs can be overlapped or orthogonal, determined by the deployment of the network operators. In consequence, the UAVs associated with different BSs may interfere with each other in the uplink transmission, as shown in Fig. 2, which is referred to as inter-cell interference. Since UAVs are likely to have line-of-sight (LoS) channels to multiple BSs due to their high altitudes, the inter-cell interference may be severe in the cellular Internet of UAVs.

### Distributed Sense-and-Send Protocol

To coordinate multiple UAVs to execute the sensing tasks in a distributed manner, we propose the following distributed sense-and-send protocol based on [8]. In this protocol, UAVs perform sensing and transmission in a synchronized iterative manner in the unit of a *sense-and-send cycle*, or *cycle* in short. In each cycle, UAVs need to sense their targets and transmit the sensory data to the BSs. As shown in Fig. 3, a cycle contains three phases: the *beaconing phase*, the sensing phase and the transmission phase, which are explained as follows.

**Beaconing Phase:** At the beginning of the beaconing phase, each BS first broadcasts a beaconing frame on the wireless control channel, which contains the identity of the BS. To synchronize the UAVs and the BSs, the synchronization signals adopted by the cellular communications can be used in the beaconing frames [9]. After receiving the beaconing frames, all the UAVs are synchronously informed that a new cycle has begun. Then the UAVs send back their state information to their associated BSs on the control channels, which includes their locations and the channel conditions toward the BSs. The BSs will exchange the state information of the UAVs with each other and then broadcast it on the wireless control channel, which can then be received by the UAVs.

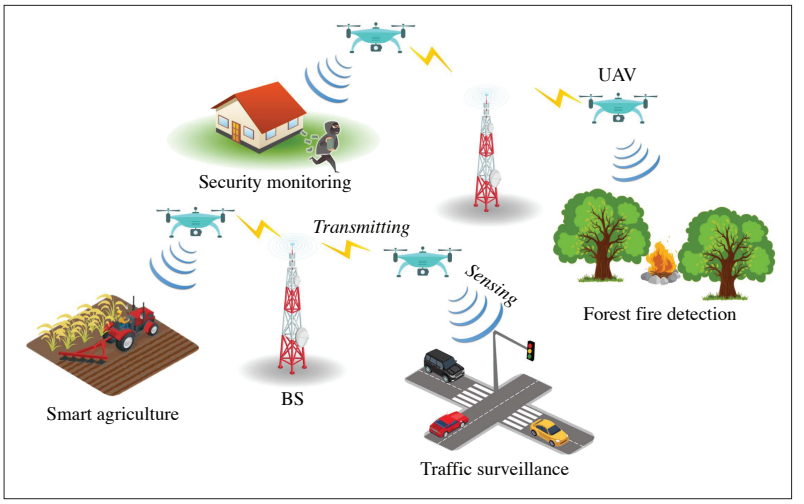


**FIGURE 1.** Cellular Internet of UAVs for various kinds of sensing tasks.

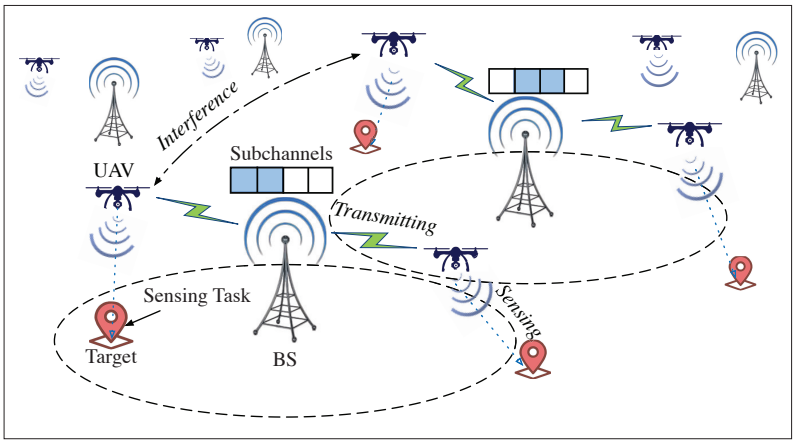


**FIGURE 2.** System model of the cellular Internet of UAVs.

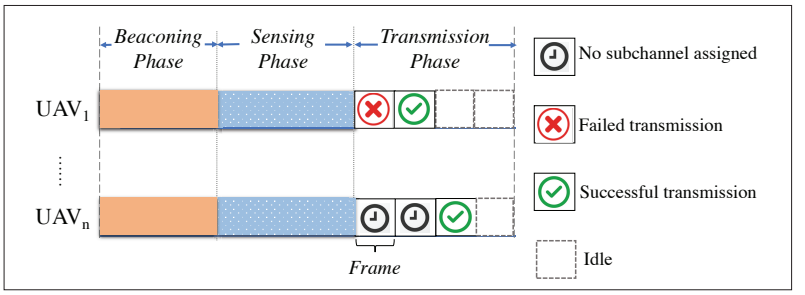


**FIGURE 3.** Sense-and-send cycle.

By this means, the BSs can obtain necessary information from the UAVs to perform subchannel allocation. Further, based on the information provided by the BSs, the UAVs can make decisions on their sensing and transmission in a distributed manner, including their user associations, trajectories, and transmit power levels. This decision making process takes place at the end of each beaconing phase. Moreover, as the duration of a cycle is pre-defined, the UAVs know when to expect the next beaconing frames from the BSs, and thus the synchronization will not be lost.

**Sensing Phase:** In the sensing phase, UAVs sense the targets of the tasks and collect sensory data. We assume that the UAVs cannot determine whether the sensing was successful or not based on their sensory data due to their limited onboard processing abilities. Therefore, the

[1] For example, the sensing is considered to be successful when the sensor successfully detects an event (e.g., a traffic jam at a crossroads) or correctly measures the condition of a target (e.g., the air quality at a certain location).

[2] In the cellular Internet of UAVs, since the UAVs are IoT devices, the uplink transmission for sensory data dominates. Therefore, in this article, we focus on the uplink transmission of the UAVs.

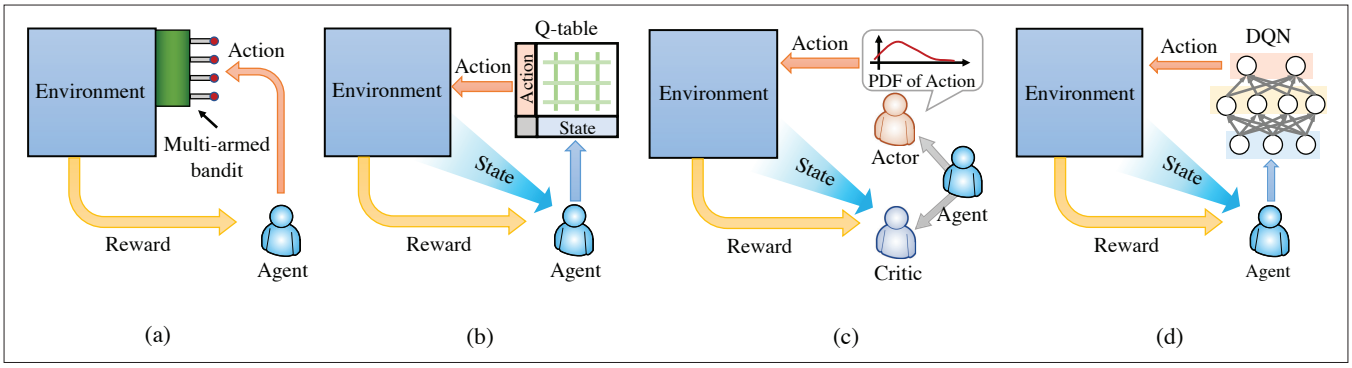


**FIGURE 4**. Illustrations of: a) multi-armed bandit learning; b) Q-learning; c) actor-critic learning; d) deep reinforcement learning.

UAVs need to send their sensory data to the BSs, and the BSs will decide whether the sensory data are valid or not.

**Transmission Phase:** In order to synchronize the transmissions of UAVs, the transmission phase of each cycle is further divided into *frames*, the basic time unit for subchannel allocation. The UAVs transmit their sensory data to the BSs through allocated subchannels in each frame. As shown in Fig. 3, for each UAV, there are four possible situations that can take place in each frame of the transmission phase:

- *No subchannel assigned:* The UAV is allocated no subchannel and needs to wait for the next frame.
- *Failed transmission:* The UAV is allocated a subchannel by the BS. However, the uplink transmission fails due to low received signal power or interference from other UAVs' transmissions. Therefore, the UAV needs to transmit again in the following frames.
- *Successful transmission:* The UAV is allocated a subchannel and transmits the sensory data to the BS successfully.
- *Idle:* The UAV keeps idle and will not transmit as it has already sent the sensory data to the BS successfully in the previous frames.

Since the spectrum resources are scarce, the subchannels of a BS may not be sufficient to support all the associated UAVs to transmit their sensory data at the same time. To handle this problem, the BSs need to adopt efficient subchannel allocation mechanisms to allocate the limited number of subchannels to the UAVs. An example of the subchannel allocation mechanisms is to allocate the subchannels to the UAVs that have the highest probabilities of successful transmission, which is adopted in [8]. Reinforcement learning can also be applied for subchannel allocation, which is discussed later.

## Reinforcement Learning for the Cellular Internet of UAVs

To better understand the applicability of reinforcement learning in the cellular Internet of UAVs, we start by introducing the basics of reinforcement learning. We then categorize reinforcement learning approaches into four types: multi-armed-bandit learning, Q-learning, actor-critic learning, and deep reinforcement learning, and give a brief introduction to each. Finally, we discuss possible applications of reinforcement learning to solve key problems in the cellular Internet of UAVs.

### Basics of Reinforcement Learning

Reinforcement learning is a learning process in which agents make decisions, observe the results, and then automatically adjust their policies to achieve their objectives [10]. To be specific, reinforcement learning is based on the Markov decision process (MDP), which consists of an environment and some agents. At each time step, the environment is at a certain state, and each agent selects a certain action according to its policy. Then the environment transits into a new state, which is determined by its previous state and the actions of the agents. A reward is generated for each agent, which quantifies how well the objective of the agent is achieved. Due to the capability of modeling state transitions, the MDP is widely applied to model sequential decision making problems in dynamic environments.

Also, in contrast to centralized scheduling approaches and supervised machine learning, reinforcement learning does not rely on accurate prior knowledge of the environment or historical labeled data. Instead, in reinforcement learning, the agents can automatically learn from the environment and their own past experiences through the rewards they obtain in order to improve their policies. This property makes reinforcement learning suitable for application in the cellular Internet of UAVs, where the UAVs face a rather dynamic and complex environment. Generally, reinforcement learning can be categorized into four types: multi-armed bandit learning, Q-learning, actor-critic learning, and deep reinforcement learning. In the following, we will elaborate on these four reinforcement learning approaches.

**Multi-Armed Bandit Learning:** As shown in Fig. 4a, in multi-armed bandit learning, the agent selects actions without recognizing the state of the environment. After each time step, a reward is received by the agent, which is relevant to the action performed in the time step. Based on the received reward for each action, the agent maintains a table of estimates of the potential rewards associated with the actions. The agent updates the potential reward estimates by a linear combination of the previous values in the table and the latest received reward. The objective of the agent is to obtain the maximum total reward, and thus the agent needs to select the current best action with the highest estimated reward. Nevertheless, as it is necessary for the agent to explore the potential reward associated with each action in order to choose the best one, there is a trade-off between *exploration* and *exploitation*. Consequently, the

agent needs to decide whether to take the current best action or search for a better one.

Since the state transition of the environment has not been considered, multi-armed bandit learning is inefficient in dealing with fast-changing environments. However, this disadvantage can be compensated for by its very low complexity in implementation due to its low memory and computation requirements. To be specific, suppose that the maximum number of available actions of the agent is $N$, and the agent only needs to store the $N$ potential rewards associated with the actions. At each time step, the agent needs to select the current best action or a random one and then update the potential reward associated with the selected action. Therefore, the computational complexity in each time step is $\mathcal{O}(N)$. In summary, multi-armed bandit learning has very low memory requirements and computational complexity.

**Q-Learning:** As shown in Fig. 4b, the agent in Q-learning selects actions based on the *Q-table*. To be specific, the value in the Q-table, that is, the *Q-value*, for each state-action pair represents the estimated total rewards for the agent under its current policy after executing the action at the state. Here, the policy of the agent is to select the action that currently has the largest Q-value at each state. To train the Q-table to be more accurate, the agent updates its Q-table based on the observed reward after each time step. Based on the updated Q-table, the policy of the agent also updates. By this means, the Q-value and the policy of the agent update iteratively, and it has been proved that the policy will eventually converge to the optimal policy of the agent [6].

To implement Q-learning, the agent who has $N$ available actions and $M$ states needs to store a Q-table involving $N \times M$ elements for all the state-action pairs. As for the computations of the agent in each time step, they are similar to those of multi-armed bandit learning, which indicates that the computational complexity is $\mathcal{O}(N)$. It can be observed that the memory requirements and computational complexity of the Q-learning are low.

**Actor-Critic Learning:** As shown in Fig. 4c, the agents in actor-critic learning are logically split into two roles, that is, a *critic* and an *actor*. The actor represents the action selection policy, which is a probability distribution function (PDF) over the action space. The critic observes the states and rewards from the environment and evaluates the state values, that is, the expected total rewards that will be received in the future passing through the states. In this sense, the critic can be considered as a state value function with respect to the state. The critic is used to improve the efficiency and stability for the training of the actor to perform optimal action selection. Specifically, after each time step, the critic updates the state value of the current state based on the observed reward. Then the actor updates its policy for the previous state based on the updated state value. Compared to other reinforcement learning algorithms, actor-critic learning does not search the action space for the action with the highest expected reward. Instead, the action is selected randomly following the PDF represented by the actor. Therefore, the complexity of action selection does not grow with the size of the action space. This characteristic makes actor-critic learning more efficient in handling large or even continuous action spaces compared to the other reinforcement learning approaches.

To implement actor-critic learning, the agent needs to store the action selection policy and the state value function. The action selection policy and state value function are both represented by parametrized functions, and thus the amount of memory to store them is determined by the number of parameters quantifying them. In each time step, the agent needs to select an action according to the policy and compute the gradients of the action selection policy and state value function with respect to the parameters in order to improve the actor and critic. In general, the action selection policy and the state value function are the combinations of basic functional elements (e.g., linear, cosine, and exponent functions) weighted by the parameters, and the number of parameters is small. Therefore, both the memory requirements and computational complexity of the actor-critic learning are low.

**Deep Reinforcement Learning:** In deep reinforcement learning, deep neural networks are utilized to handle high-dimensional state spaces [11]. As shown in Fig. 4d, at each time step, the agent inputs the feature vector of the current state into the deep neural network, which can estimate the Q-value for each action, and thus is referred to as a deep Q-network (DQN). The agent then selects the action with the largest estimated Q-value, and stores the experience, including the state transition and the reward, into a replay buffer, which is used to train the DQN to estimate Q-values more accurately.

To implement deep reinforcement learning, the agent needs to store the DQN and a replay buffer that stores its previous experiences. Generally, to obtain better results, the sizes of the DQN and the replay buffer need to be large, which results in a high memory requirement. The training of the DQN requires the gradients of estimated Q-values with respect to the parameters of the DQN, which leads to high computation complexity given a large DQN. Therefore, deep reinforcement learning requires more memory and higher computational complexity than other reinforcement learning approaches. Nevertheless, to alleviate the high memory and computational requirements on the agent, the DQN can be trained in an offline manner. For example, an agent can upload its experiences to a server. The server trains the DQN according to the experiences and returns the updated DQN to the agent periodically.

## Applications in the Cellular Internet of UAVs

Due to the rapid development of UAVs, UAVs are able to have enough onboard computation and memory capacities to perform reinforcement learning approaches, either by having in-built circuits or carrying additional computing devices. This allows the implementations of reinforcement learning approaches to solve key problems in the cellular Internet of UAVs. In the following, we discuss the possible implementations of reinforcement learning to solve the trajectory control and resource management problems in the cellular Internet of UAVs, including user association, power management, and subchannel allocation.

**Multi-Armed Bandit Learning for User Association:** In the cellular Internet of UAVs, each

Due to the rapid development of UAVs, UAVs are able to have enough onboard computation and memory capacities to perform reinforcement learning approaches, by either having built-incircuits or carrying additional computing devices. This allows the implementations of reinforcement learning approaches to solve key problems in the cellular Internet of UAVs.

The onboard batteries of UAVs are generally very limited; therefore, high transmit power results in the batteries draining quickly. In order to maximize the total number of successful transmissions before the battery runs out, it is crucial for each UAV to adopt efficient approaches for transmit power management.

BS is equipped with $K$ subchannels to support at most $K$ UAVs to transmit simultaneously, and each UAV needs to choose one BS with which to associate. From the protocol proposed above, it can be observed that the probabilities of successful transmission for UAVs will decrease with the number of UAVs associated with the same BS due to the aggravated competition for the subchannels. Besides, due to the shadowing in the air-to-ground channels, the channel conditions from a UAV to different BSs are usually varying. For example, when the channel between the UAV and the BS has an LoS component, its path loss is much lower than when no LoS component exists [12]. For the above reasons, the user association is a challenging problem in the cellular Internet of UAVs.

Since multi-armed bandit learning does not rely on prior information such as the channel conditions between the UAVs and the BSs, it is suitable for the user association problem. In this problem, each UAV can be considered as an agent. At the beginning of each beaconing phase, each UAV decides which BS with which to associate. As the objective of the UAV is to maximize the successful transmission probability, the reward is 1 if the sensory data is successfully transmitted; otherwise, the reward is 0. Each UAV optimizes its action selection policy by estimating the expected reward for each action and selecting the action with the highest expected reward. Further, in order to keep searching for a better action, the UAV has an exploration probability, with which it selects an action randomly.

**Q-Learning for Trajectory Control:** When the UAV executes a sensing task, the successful sensing probability will be higher if it gets closer to the sensing target. On the other hand, the UAV will have a higher probability of transmitting the sensory data to the BS successfully if it approaches the BS. Therefore, sensing and transmission are coupled with its trajectory. However, since accurate sensing and transmission models are hard to obtain in the complex and dynamic environment of the cellular Internet of UAVs, trajectory control is also challenging.

As Q-learning does not require models of the sensing and transmission, it is suitable for the trajectory control problem. For applying Q-learning, the flight space can be abstracted into a finite set of discrete spatial points, and the trajectory can be considered as a path through these spatial points. The state of each UAV is its location, and the action is its trajectory in each cycle. Since the objective of each UAV is to send valid sensory data to the BS, the reward is 1 if the valid sensory data is received successfully by the BS; otherwise, the reward is 0.

**Actor-Critic Learning for Power Management:** In the cellular Internet of UAVs, the UAVs need to raise their transmit power in order to increase the successful transmission probability and satisfy the QoS requirement. However, the onboard batteries of UAVs are generally very limited; therefore, high transmit power results in the batteries draining quickly. In order to maximize the total number of successful transmissions before the battery runs out, it is crucial for each UAV to adopt efficient approaches for transmit power management.

Since the transmit power level can take values from a continuous set, it is suitable to apply actor-critic learning for the power management problem. To be specific, in each cycle, the path loss of the uplink subchannel and remaining battery capacity can be jointly considered as the state. At the end of the beaconing phase, the actor of the UAV selects a transmit power level according to the current state. After the cycle, the reward to the UAV is 1 if the transmission was successful; otherwise, the reward is 0. The experience, consisting of the state transition and the reward, is used to train the critic to estimate the value of the state more accurately. Then the actor updates itself to select a better action by using the critic's evaluation.

**Deep Reinforcement Learning for Subchannel Allocation:** In the cellular Internet of UAVs, the uplink transmissions of the UAVs may suffer severe inter-cell interference due to the low path loss of LoS channels between each UAV and multiple BSs. To alleviate the inter-cell interference and improve the probability of successful transmission, the BSs that share the same frequency band need to perform subchannel allocation jointly.

However, in the subchannel allocation, the channel conditions between multiple BSs and their associated UAVs need to be taken into consideration. Moreover, since the UAVs that are idle in the frame do not transmit, the transmission states of all UAVs should also be considered. Therefore, the subchannel allocation problem has a complex and high-dimensional state space. As deep reinforcement learning is able to solve the optimal policies for agents facing a high-dimensional state space, it is suitable for the subchannel allocation problem in the cellular Internet of UAVs.

In this case, the BSs that share a certain frequency band can be considered as the agent, and the action of the BSs is the subchannel allocation for the UAVs. In each frame in the transmission phase, the state is composed of the following elements:

- The path loss of the channels between the involved BSs and their associated UAVs
- The indicators of whether the UAVs are idle in the frame

At each frame in the transmission phase, the BSs form a feature vector representing the current state and input it into the DQN, which returns the Q-value for each possible subchannel allocation of the UAVs. Then the BSs select the subchannel allocation with the largest Q-value as their action. After the cycle, the reward given to the agent can be designed as the number of successful transmissions. The rewards, along with the transition among states, are stored in the replay buffer of the UAV, which is then used to train the DQN to estimate the Q-values for the state-action pairs more accurately.

In Table 1, we summarize the different types of reinforcement learning approaches with their characteristics and their applications in the cellular Internet of UAVs.

## Reinforcement Learning Example: Q-Learning for Trajectory Control

As an illustrative example, in this section, we introduce how to apply the reinforcement learning approach to solve the trajectory control problem in the cellular Internet of UAVs.

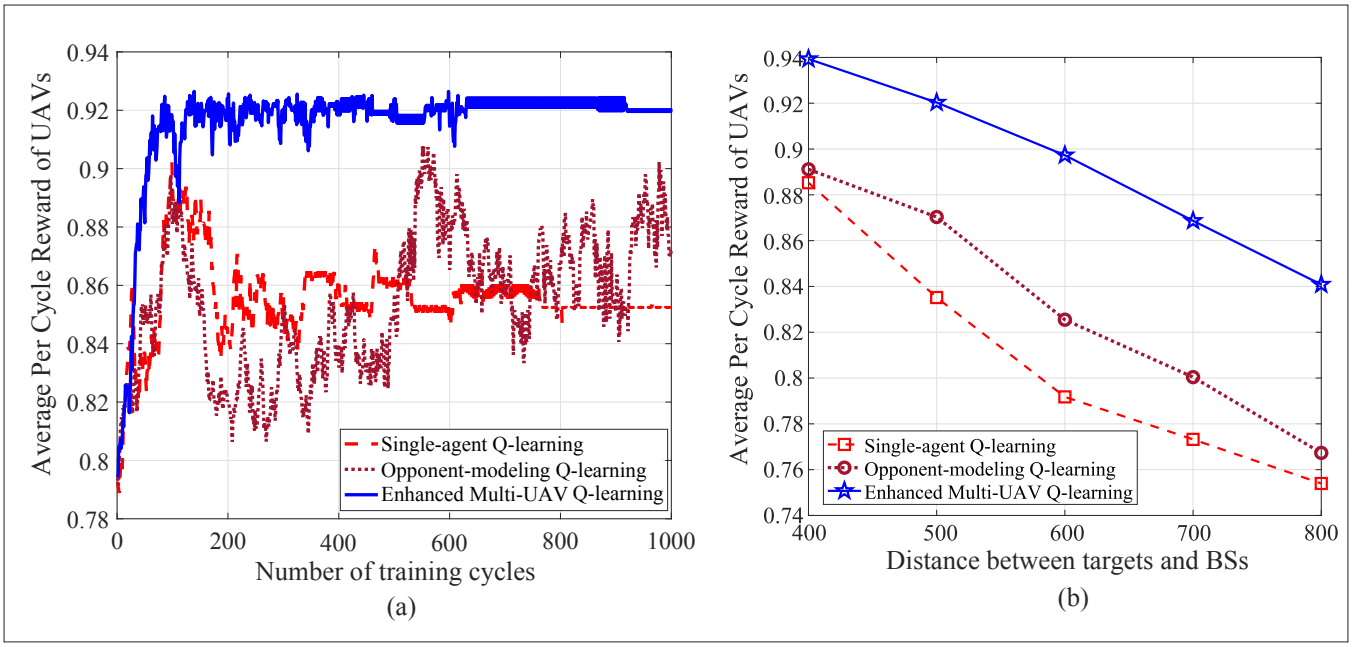


**FIGURE 5**. UAVs' average per cycle reward vs.: a) number of training cycles; b) distance between the targets and the BSs.

## Problem Description

We consider a cellular Internet of UAVs, which consists of two BSs and three UAVs associated with each BS. The two BSs are assumed to have two subchannels to support the uplink transmissions of the UAVs, and the frequency bands utilized by the two BSs are different. Moreover, the UAVs are flying in a cylindrical space centered at the BS, with minimum and maximum flying altitude constraints.

To depict the trajectory, we divide the space into a finite set of discrete spatial points, which are arranged in a square lattice pattern. The trajectory control problem can be reformulated as that in each cycle. Each UAV selects an adjacent spatial point of its current location which can maximize the total rewards, that is, the number of valid sensory data elements successfully sent to the associated BS. The distance from the current position to the next one should be less than the UAVs' maximum flying distance within one cycle. Note that for each UAV, the valid sensory data received by the BS in the far future will be worth less than those received soon, which means the UAVs have discount values on their future rewards.

## Algorithm Design

To solve the trajectory control problem, we utilize the Q-learning algorithm where the UAVs are considered as agents, the locations of UAVs are the state, and the cycle is the time step. The available action set for a UAV consists of all the possible direct trajectories to the feasible spatial points whose distance is less than the maximum flying distance of a UAV within one cycle. The state transition is the mapping from the current locations and actions of UAVs to the locations of the UAVs at the beginning of the next cycle. Finally, the reward is 1 if the BS receives valid sensory data from the UAV; otherwise, the reward is 0.

To be specific, the Q-learning algorithm for the trajectory control problem can be briefly described as follows:

- At the beginning of a cycle, each UAV chooses the action that has the maximum Q-value in the current state.
- The UAV performs the selected action in the cycle.
- At the beginning of the next state, the UAV observes the transition state and is informed whether valid sensory data has been received by the BS.
- The UAV updates the state-action value in the previous state, and then selects an action for the new state.

Moreover, to improve the efficiency of the Q-learning for trajectory control, we propose an enhanced multi-UAV Q-learning algorithm in [8], which is based on the following observations. First, since all the UAVs determine trajectories at the same time, each UAV needs to consider other UAVs' action selections when selecting its own. This can be achieved by each UAV keeping a record of other UAVs' action selection statistics in each state [13].

Second, it can be observed that if the UAV does not locate near the vertical plane passing through the BS and the target, both the successful sensing and transmission probabilities will decrease. Therefore, the available action set of each UAV can be reduced to the trajectories toward or on the BS-target plane.

Third, in the traditional Q-learning algorithms, agents update their values using the observed reward in the last cycle. However, in such a manner, the estimated Q-values converge slowly, and the performance of the algorithms are likely to be poor. Therefore, in the proposed algorithm, UAVs update their Q-functions based on the probability of successful valid sensory data transmission, which can be calculated by the algorithm proposed in [8].

To evaluate the performance of the proposed enhanced multi-UAV Q-learning algorithm, we compare it to the traditional single-agent and opponent modeling Q-learning algorithms [13]. Figure 5a shows the average per cycle reward of

As a powerful approach for UAVs to automatically learn optimal decision-making policies from past experiences, reinforcement learning is promising for the cellular Internet of UAVs. Nevertheless, there are still many open issues in this field, which may drive future research. Several potential research directions are listed below as examples.

the UAVs vs. the number of training cycles. It can be seen that compared to the traditional Q-learning algorithms, the proposed algorithm converges faster and to a higher reward. Figure 5b shows the average per cycle reward of the UAVs vs. the distance between the targets and the BSs. It indicates that the probability of successful valid data transmission decreases with the distance between the targets and the BSs. Nevertheless, the decrement in the proposed algorithm is less than those in the other algorithms. This indicates that the proposed algorithm is more robust to the variance of the targets' locations.

## Conclusions and Future Outlook

In this article, reinforcement learning has been introduced as a distributed approach to solve key problems in the cellular Internet of UAVs. We have introduced the cellular Internet of UAVs and then proposed a distributed sense-and-send protocol for the coordination of multiple UAVs to execute sensing tasks. Following that, we have introduced the basics of reinforcement learning and the four types of reinforcement learning approaches. We have also discussed the potential applications of reinforcement learning approaches to tackle the trajectory control and resource management problems in the cellular Internet of UAVs. To provide an example, we have elaborated on using the enhanced multi-UAV Q-learning algorithm to solve the trajectory control problem.

As a powerful approach for UAVs to automatically learn optimal decision making policies from past experiences, reinforcement learning is promising for the cellular Internet of UAVs. Nevertheless, there are still many open issues in this field, which may drive future research. Several potential research directions are listed below as examples.

### Cooperative Cellular Internet of UAVs

When the targets are far away from the coverage of the BSs, the UAVs may need cooperation to execute the tasks. To be specific, a UAV can choose not to sense any targets, but to work as a relay that helps another UAV transmit sensory data to the BSs [14]. In this case, the UAVs need to select their roles in each cycle, with the objective of maximizing the total number of valid sensory data received by the BS. To tackle this problem, Q-learning can be applied. Specifically, the state can be the locations of the UAVs and the BSs, and the actions of the UAVs are their decisions on whether to sense or to relay. Moreover, in accordance with the objective of the UAVs, the rewards can be defined as the number of valid sensory data elements received by the BS in each cycle.

### Cognitive Cellular Internet of UAVs

In some sensing tasks (e.g., live streaming), the transmission of a large amount of sensory data generated by UAVs may pose a significant burden on cellular networks. To guarantee the QoS for traditional cellular users while improving the QoS of data transmission for UAVs, cognitive radio can be used to enable the UAVs to opportunistically access the channels nominally occupied by cellular users. Under such a setting, cellular users and UAVs serve as the primary and secondary users, respectively. This channel access problem can be solved by deep reinforcement learning [15], where UAVs are the agents, and the previous observations on the subchannels are regarded as the state. To avoid interfering with primary users, the rewards of UAVs can be designed as weighted sums of successful transmission probabilities and the interference caused to the primary users.

### Millimeter-Wave Cellular Internet of UAV

In the cellular Internet of UAVs, sensory data need to be transmitted to the BSs in a timely manner, which may require high data rates between the UAVs and the BSs. Therefore, it is promising to apply millimeter-wave (mmWave) communication in the cellular Internet of UAVs, which can provide abundant frequency spectrum resources and alleviate the inter-cell interference due to its high attenuation rate. To implement mmWave communication, beamforming is required at UAVs to steer strong signal-to-noise ratio (SNR) at the BS, in which the UAVs need to search a large number of beam directions and find the best angle. To solve this problem, actor-critic learning can be adopted. Specifically, the locations of UAVs and BSs can be jointly considered as the states, and the beam directions can be considered as the UAVs' action space. In each cycle, the UAV selects a beam direction according to the PDF generated by its actor role. The rewards for UAVs can be designed as the received SNR level at the BSs in the cycle, which is in accordance with the objective of finding the optimal beam direction.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under grant number 61625101 and in part by U.S. AFOSR MURI 18RT0073 and MURI FA9550-18-1-0502, NSF EARS-1839818, CNS1717454, CNS-1731424, CCF-1908308, and CNS-1646607.

## References

[1] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint," *IEEE Commun. Surveys & Tutorials*, vol. 18, no. 4, Apr. 2016, pp. 2624–64.
[2] J. Wang *et al.*, "Taking Drones to the Next Level: Cooperative Distributed Unmanned-Aerial Vehicular Networks for Small and Mini Drones," *IEEE Vehic. Tech. Mag.*, vol. 12, no. 3, July 2017, pp. 73–82.
[3] H. Zhang *et al.*, "Cooperation Techniques for a Cellular Internet of Unmanned Aerial Vehicles," *IEEE Wireless Commun.*, vol. 26, no. 5, Oct. 2019, pp. 167–73.
[4] S. Zhang *et al.*, "Cellular Cooperative Unmanned Aerial Vehicle Networks with Sense-And-Send Protocol," *IEEE Internet of Things J.*, vol. 6, no. 2, Apr. 2019, pp. 1754–67.
[5] M. Mozaffari *et al.*, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Commun. Surveys & Tutorials*, vol. 21, no. 3, Mar. 2019, pp. 2334–60.
[6] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.
[7] V. V. Shakhov and I. Koo, "Experiment Design for Parameter Estimation in Probabilistic Sensing Models," *IEEE Sensors J.*, vol. 17, no. 24, Dec. 2017, pp. 8431–37.
[8] 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 J.*, vol. 6, no. 4, Oct. 2018, pp. 6177–89.
[9] A. Fotouhi *et al.*, "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," *IEEE Commun. Surveys & Tutorials*, to be published.
[10] E. Alpaydm, *Introduction to Machine Learning*, 3rd ed., MIT Press, 2014.

[11] K. Arulkumaran *et al.*, "Deep Reinforcement Learning: A Brief Survey," *IEEE Signal Processing Mag.*, vol. 34, no. 6, Nov. 2017, pp. 26–38.
[12] 3GPP, "Enhanced LTE Supported for Aerial Vehicles," TR 36.777, May 2017.
[13] C. Claus and C. Boutilier, "The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems," *Proc. IAAI*, Madison, WI, July 1998.
[14] S. Zhang *et al.*, "Joint Trajectory and Power Optimization for UAV Relay Networks," *IEEE Commun. Lett.*, vol. 22, no. 1, Jan. 2018, pp. 161–64.
[15] S. Wang et al., "Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks," *IEEE Trans. Cogn. Commun. Net.*, vol. 4, no. 2, Feb. 2018, pp. 257–65.

## Biographies

Jingzhi Hu [S'16] (jingzhi.hu@pku.edu.cn) is a Ph.D. student at the School of Electrical Engineering and Computer Science, Peking University, China. He received his B.S. degree in electronic engineering from Peking University in 2017. His current research interests include reinforcement learning, compressive sensing, and wireless network virtualization.

Hongliang Zhang [S'15, M'19] (hongliang.zhang92@gmail.com) received his B.S. and Ph.D. degrees at the School of Electrical Engineering and Computer Science, Peking University, in 2014 and 2019, respectively. Currently, he is a postdoctoral fellow in the Electrical and Computer Engineering Department as well as the Computer Science Department at the University of Houston, Texas. His current research interests include cooperative communications, Internet of Things networks, hypergraph theory, and optimization theory. He has also served as a TPC member for IEEE GLOBECOM 2016, ICC 2016, ICCC 2017, ICC 2018, GLOBECOM 2018, ICCC 2019, and GLOBECOM 2019. He is currently an Associate Editor for *IET Communications*.

Lingyang Song [S'03, M'06, SM'12, F'19] (lingyang.song@pku.edu.cn) received his Ph.D. from the the University of York, United Kingdom, in 2007, where he received the K. M. Stott Prize for excellent research. He worked as a research fellow at the University of Oslo, Norway, until rejoining Philips Research UK in March 2008. In May 2009, he joined the Department of Electronics, School of Electronics Engineering and Computer Science, Peking University, and is now a Boya Distinguished Professor. His main research interests include wireless communication and networks, signal processing, and machine learning. He was the recipient of the IEEE Leonard G. Abraham Prize in 2016 and the IEEE Asia Pacific Young Researcher Award in 2012. He has been an IEEE Distinguished Lecturer since 2015. He is a Clarivate Analytics Highly Cited Researcher.

Zhu Han [S'01, M'04, SM'09, F'14] (zhan2@uh.edu) received his B.S. degree in electronic engineering from Tsinghua University in 1997, and his M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively. From 2000 to 2002, he was an R&D engineer at JDSU, Germantown, Maryland. From 2003 to 2006, he was a research associate at the University of Maryland. From 2006 to 2008, he was an assistant professor at Boise State University, Idaho. Currently, he is a John and Rebecca Moores Professor in the Electrical and Computer Engineering Department as well as in the Computer Science Department at the University of Houston. He is also a Chair professor at National Chiao Tung University, Republic of China. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He received an NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing* in 2015, the IEEE Leonard G. Abraham Prize in the field of Communications Systems (best paper award in *IEEE JSAC*) in 2016, and several best paper awards at IEEE conferences. He was an IEEE Communications Society Distinguished Lecturer from 2015 to 2018, and is an AAAS fellow since 2019 and ACM Distinguished Member since 2019. He has been a 1 percent highly cited researcher since 2017 according to Web of Science.

H. Vincent Poor [S'72, M'77, SM'82, F'87] (poor@princeton.edu) is the Michael Henry Strater University Professor at Princeton University. He received his Ph.D. in EECS from Princeton in 1977, and from then until joining the Princeton faculty in 1990, he was on the faculty of the University of Illinois at Urbana-Champaign. During 2006–2016, he served as Dean of Princeton's School of Engineering and Applied Science. He has also held visiting positions at several other universities, including most recently at Berkeley and Cambridge. His research interests are in the areas of information theory and signal processing, and their applications in wireless networks, energy systems, and related fields. He is a member of the U.S. National Academy of Engineering and the U.S. National Academy of Sciences, and is a foreign member of the Chinese Academy of Sciences, the Royal Society, and other national and international academies. Recent recognition of his work includes the 2017 IEEE Alexander Graham Bell Medal, the 2019 ASEE Benjamin Garver Lamme Award, and honorary doctorates from Syracuse University and the University of Waterloo, awarded in 2017 and 2019, respectively.

To implement mmWave communication, beamforming is required at UAVs to steer strong SNR at the BS, in which the UAVs need to search a large number of beam directions and find the best angle. To solve this problem, actor-critic learning can be adopted.