Dynamic Spectrum Interaction of UAV Flight Formation Communication With Priority: A Deep Reinforcement Learning Approach

Yun Lin, *Member, IEEE*, Meiyu Wang, Xianglong Zhou, Guoru Ding, *Senior Member, IEEE*, and Shiwen Mao, *Fellow, IEEE*

***Abstract*—The formation flights of multiple unmanned aerial vehicles (UAV) can improve the success probability of single-machine. Dynamic spectrum interaction solves the problem of the ordered communication of multiple UAVs with limited bandwidth via spectrum interaction between UAVs. By introducing reinforcement learning algorithm, UAVs can continuously obtain the optimal strategy by continuously interacting with the environment. In this paper, two types of UAV formation communication methods are studied. One method allows for information sharing between two UAVs in the same time slot. The other method is the adoption of a dynamic time slot allocation scheme to complete the alternate use of time slots by the UAV to realize information sharing. The quality of experience (QoE) is introduced to evaluate the results of UAV sharing, and the M/G/1 queuing model is used for priority and to evaluate the packet loss of UAV. In terms of algorithms, a combination of deep reinforcement learning (DRL) and the long-short-term memory (LSTM) network is adopted to accelerate the convergence speed of the algorithm. The experimental results show that, compared with the Q-learning and deep Q-network (DQN) methods, the proposed method achieves faster convergence and better performance with respect to the throughput rate.**

***Index Terms*—Multi-unmanned aerial vehicles (UAV), self-determination, quality of experience(QoE), M/G/1 queuing model, deep reinforcement learning (DRL), long-short-term memory (LSTM).**

I. Introduction

THE SPECTRUM interaction technology of the flight formations of unmanned aerial vehicles (UAVs) is implemented to solve the spectrum sharing problem between UAVs. Because the bandwidth required for UAV information transmission is relatively large, this paper explores how to maximize the spectrum utilization in a limited frequency band. The mission requirements of UAVs are becoming increasingly greater. A UAV may often face multiple tasks, but the payload of the drone makes it impossible to carry all the combat modules in one flight. This requires multiple UAVs to form a flight formation to work together. Additionally, when a UAV performs tasks in the no-man's land, it cannot communicate with the ground station in sufficient time, thus, information sharing between UAVs is particularly important [1].

Information sharing between drones allows them to better accomplish tasks, while also ensuring the survivability of missions in the wild. Information sharing between drones can be categorized into two situations: (1) information exchange between two UAVs in each time slot, which requires the division of channel usage between UAVs and assigns the UAVs that interact in the same time slot; (2) in each time slot, one UAV transmits information in the form of a broadcast, and the remaining drones receive information; this requires dynamic allocation of time slots to ensure that only one UAV is used for information transmission, and that the other UAVs are in the receiving state [2], [3].

Reinforcement learning (RL), is a research hot spot in the field of machine learning, and has been widely used in industrial manufacturing [4], simulations [5], robot control [6], optimization and scheduling [7], games [8] and other fields. The basic idea of RL is to learn the optimal strategy for accomplishing the goal by maximizing the cumulative reward value obtained by the agent from the environment [9]. Therefore, the RL method is more focused on learning strategies to solve problems. With the rapid development of human society, and with the necessity of completing increasingly more complex real-world tasks, it is necessary to use RL to automatically learn the abstract representation of large-scale input data, and to use this characterization as a self-incentive RL to optimize the problem-solving strategy.

Google's artificial intelligence research team Deep Mind has innovatively combined the sensible DL with the decision-making RL to create a new popular research topic in the field of artificial intelligence, namely deep reinforcement learning (DRL). Since its development, the Deep Mind team has constructed and implemented human expert-level agents for use in many challenging situations. These agents build and learn

Manuscript received October 24, 2019; revised January 11, 2020; accepted January 31, 2020. Date of publication February 12, 2020; date of current version September 9, 2020. This work is supported in part by the National Natural Science Foundation of China (61771154) and the Fundamental Research Funds for the Central Universities (HEUCFG201830). This paper is also funded by the International Exchange Program of Harbin Engineering University for Innovation-oriented Talents Cultivation. The associate editor coordinating the review of this article and approving it for publication was J. Wen. *(Corresponding author: Yun Lin.)*

Yun Lin, Meiyu Wang, and Xianglong Zhou are with the College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China (e-mail: linyun@hrbeu.edu.cn; hrbeumeiyu@hrbeu.edu.cn; zhouxl@hrbeu.edu.cn).

Guoru Ding is with the College of Communications Engineering, Army Engineering University of PLA, Nanjing 210007, China (e-mail: dr.guoru.ding@ieee.org).

Shiwen Mao is with the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849 USA (e-mail: smao@ieee.org).

Digital Object Identifier 10.1109/TCCN.2020.2973376

their own knowledge directly from the original input signal without any manual coding or domain knowledge [10].

Boeing Company used the X-45 drone to conduct a two-aircraft formation flight for the first time [11]. The Royal Air Force completed the implementation of a simulated attack on ground moving targets by commanding a modified "Surf" fighter-bomber to command three UAV "simulators". The United States successfully conducted a drone aerial refueling test using the Global Hawk UAV, marking a breakthrough in drone flight and coordination [12].

Gao *et al.* [13] proposed a DSA algorithm for LTE cellular systems. It combines distributed enhanced learning and standardized inter-cell interference coordination signaling in the LTE downlink. Ni *et al.* [14], proposed a channel spectrum access algorithm based on online synchronous Q-learning to avoid channel congestion in cognitive radio networks. A non-cooperative shared spectrum allocation algorithm for multi-user and multi-channel cognitive radio systems is proposed in the research of multi-agent allocation [15]. A special recurrent neural network using reservoir calculation has been found to improve the slow convergence and learning efficiency of Q-learning [16]. To improve the network performance of multi-hop cognitive radio networks, two enhanced learning methods have been proposed for sensing and access [17]. Compared to traditional reinforcement learning methods, the number of sensors used for sensing is reduced, and throughput and energy efficiency are improved [18]. The combination of reinforcement learning algorithms and game theory can help to better solve problems in complex environments. Lundén *et al.* [19] proposed a spectrum access algorithm that can better accommodate channel assignment for multi-agent collaborative work. For UAVs, the algorithm should have fast computational power to ensure that the best strategy can be quickly calculated.

The priority mechanism ensures that important information is preserved in the event of a crowded communication environment. The mission modules carried by each UAV in the UAV formation generally do not perform the same important tasks. The importance of the mission either light and or heavy, and some module information may be insignificant for task decision-making. However, some module information is the core of the entire task, so the priority mechanism can ensure that important information can be shared in time and non-important information can be queued and eventually discarded according to the queuing model, so that it generally does not have much impact on the success of the task execution.

In this paper, an intelligent decision-making UAV formation information sharing mechanism that uses reinforcement learning without any prior knowledge is designed. The interaction between the strategies is learned and generated by the communication of the UAV with the environment and other UAVs.

The main contributions of this paper are as follows:

- All allocation strategies are based on a priority mechanism. Priority is used to classify the importance of the UAV task to ensure that the data collected by important modules in the UAV formation can be shared in time to avoid loss. For the actual situation of UAV formation, the M/G/1 queuing model is used for priority determination and packet loss determination.
- According to the delay of the communication system, the packet loss rate and other information, the reward function of the reinforcement learning is redesigned in combination with quality of service (QoS). The parameter variable of the communication system is added to design the reward function as the mean opinion score (MOS), which is closer to the real communication system.
- The algorithm of deep reinforcement learning (DRL) combined with the long short-term memory network (LSTM) is proposed. It is found to have a better convergence speed and better performance than the traditional deep Q-network (DQN) algorithm.

The experimental results show that compared with the traditional reinforcement learning methods (Q-learning and the deep Q-network (DQN)), the proposed method achieves faster convergence and better performance with respect to average collision rate, MOS, and throughput. The remainder of the paper is organized as follows. The system model and queuing model are introduced in Section II, and the proposed dynamic channel and dynamic time slots allocation schemes are introduced in Section III. The performance of the proposed scheme is evaluated in Section IV, and the paper is concluded in Section V.

II. Preliminary Knowledge and System Model

There are many advantages to flying a UAV. Multiple UAV formation flights, coordinated reconnaissance, and combat modes can improve the success probability of single-machine single combat missions to a certain extent. The main research object in this paper is the dynamic management of information exchange in the cooperative operation of UAV formation. The communication link of UAV studied in this paper belongs to the downlink of UAV, that is to use the downlink of UAV for information transmission between UAVs. The difference between dynamic channel allocation and dynamic slot allocation is mainly reflected in the communication mode of UAV, the former is mainly single channel communication, and the latter belongs to broadcast communication mode. The layer between UAVs is reflected by priority.

A. UAV Formation Model

The typical application scenario in which multiple UAVs are deployed is considered in this work, and is illustrated in Fig. 1. Five UAV flight formations are designed, and the priority mechanism is adopted. One UAV is used as the highest priority to serve as the temporary command decision center. The layered structure can ensure that the high priority UAV takes up the resources first, and prevent the low priority UAV from taking up the resources and not using them, resulting in the waste of spectrum.

- *Dynamic channel model* [37]–[39]. The UAV is limited by the communication distance of the ground command center. When the unmanned area performs the task, it can only rely on mutual information exchange to

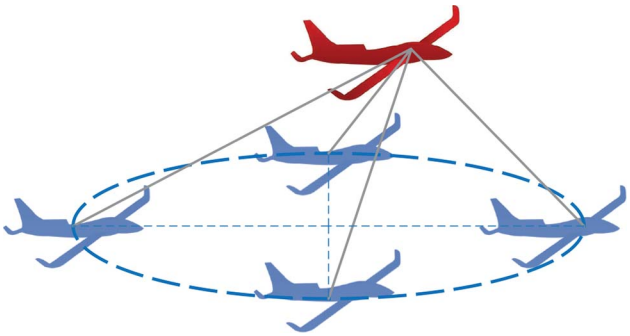

Fig. 1. System environment model. An example of a formation of 5 UAVs sharing information under the layered architecture.

Fig. 2. Time slot model. At the front of each time slot, the current state is first evaluated, the action to be taken (the UAV number to perform information sharing) is then selected, the information is transmitted, and finally the policy is updated at the end of the round.

keep abreast of the task execution level in the current environment and the next execution plan. The dynamic channel model is used to solve the process of information exchange between two UAVs, and the channel is divided in advance. In the same time slot, two UAVs share one channel to complete information sharing. Information sharing includes information transmission and receiving. The information sharing between the UAVs is completed by the reinforcement learning algorithm.

- *Dynamic time slot model* [40]–[41]. The dynamic slot model primarily solves how to ensure that different priority UAVs occupy time slots for information sharing. The sharing of information to all other UAVs in the shared state of each UAV achieves the purpose of information sharing. The task assignment under each time slot is shown in Fig. 2.

B. Queuing Model for Packet Loss Determination

Reinforcement learning is a Markov decision process. This paper applies the generalized Markov update process to establish a finite population M/G/1 queuing model [20]. The queuing system has four main features: a limited system customer source, the single service desk, the obtained service customer returns to the customer overall, and the strength of the customer arrival flow depends on the state of the system itself. This is consistent with the network model of the UAV formation system [26]. Therefore, the M/G/1 queuing model is adopted to measure the packet loss rate and service status of the UAV. The priority design of UAV formation based on the queuing model ensures the timely sharing of important information [27].

The principle followed is that a high-priority UAV can interrupt the channel or time slot occupancy of a low-priority UAV, but a low-priority UAV cannot interrupt the occupation

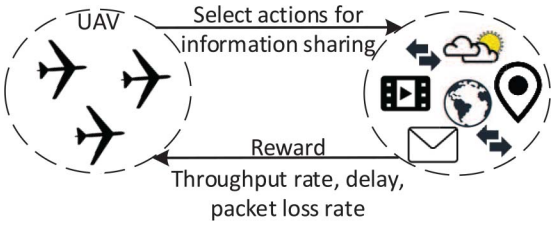

Fig. 3. The MDP model of UAV formation. The UAVs choose the action for information sharing, and environment feedback includes status (throughput rate, delay, and packet loss rate) and rewards.

of a high-priority UAV [28]. In the channel dynamic allocation scheme, once the low-priority UAV is interrupted, it may choose to wait for the appropriate time to continue to occupy the channel for sharing, or switch to another channel to share information with another UAV [29]. In the dynamic time slot allocation scheme, the interrupted low-priority UAV can only choose to wait for the high-priority UAV to occupy the time slot before sharing information, and the UAV packet loss situation and delay can be determined according to the queuing theory model [30].

C. Deep Reinforcement Learning

In the multi-UAV environment, it is challenging to obtain a large number of data samples for training, and the training process is time consuming and computation intensive. To this end, reinforcement learning (RL) provides an effective solution.

RL is a type of learning from environmental state mapping to action, with the goal of maximizing the cumulative reward for agents in their interactions with the environment [21]. When interacting with the environment, the RL agents record their status, actions, and goals. A state is a specific configuration of an environment that is sensed by its agents and affected by its operations [31]. The goal of the agent is to maximize the user-defined reward amount received from the environment as its behavioral feedback. For example, negative rewards can motivate agents to "adjust" their behavior to have a higher reward. The total reward for all possible future returns of an action is defined as the reward value. These concepts are all formalized in the MDP framework illustrated in Fig. 3.

The Markov decision process (MDP) can be used to model RL problems. MDP is usually defined as a four-tuple (S, A, ρ, f), where [10]:

- S is a collection of all environmental states, $s_t \in S$ represents the state of the *agent* at time t;
- A is the set of executable actions of the agent, $a_t \in A$ is the action taken by the *agent* at time t;
- ρ: $S \times A \to R$ is the reward function, $\gamma \sim \rho(s, a)$ represents the immediate reward value obtained by the *agent* performing action a_t in state s_t;
- f: $S \times A \times R \to [0, 1]$ is the state transition probability distribution function, $s_{t+1} \sim f(s_t, a_t)$ represents the probability that the agent will perform the action a_t transition to the next state s_{t+1} in state s_t.

In RL, the policy π: $S \to R$ is a mapping of the state space to the action space. It is expressed as the agent selecting action a_t in state s_t performing the action and propositioning to the

next state s with probability $f(s, a)$, while accepting rewards r_t from environmental feedback. Assuming that the immediate reward for each time step in the future must be multiplied by a discount factor γ, then from the time t to the end of the time T, the sum of the rewards is defined as [15]:

$$R_t = \sum_{t'=t}^{T} \gamma^{t'-t} \gamma_{t'}, \tag{1}$$

where $\gamma \in [0, 1]$, and it is used to weigh the impact of future rewards on cumulative rewards.

The state action value function $Q(s, a)$ refers to the execution of action A in the current state S, and always follows the strategy to the end of the episode, in which the cumulative return obtained by the agent is expressed as [15]:

$$Q^{\pi}(s, a) = E[R_t | s_t = s, a_t = a, \pi]. \tag{2}$$

For all state action pairs, if the expected return of one policy π^* is greater than or equal to the expected return of all other strategies, the policy π^* is called the optimal policy. There may be more than one optimal strategy, but they share a state action value function [15]:

$$Q^*(s, a) = \max_{\pi} E[R_t | s_t = s, a_t = a, \pi] \tag{3}$$

this is called the optimal state-action value function, which follows the Bellman optimal equation:

$$Q^*(s, a) = E_{s'-S}\left[r + \gamma \max_{a'} Q\left(s', a'\right) | s, a \right]. \tag{4}$$

In traditional RL, the Q-value function is generally solved by the iterative Bellman equation:

$$Q_{i+1}(s, a) = E_{s'-S}\left[r + \gamma \max_{a'} Q\left(s', a'\right) | s, a \right], \tag{5}$$

in which, if $i \to \infty$, then $Q_i \to Q^*$. In other words, by continuously iterating, the state-action value function will finally converge, and the optimal strategy will be obtained: $\pi^* = \text{argmax}_{a \in A} Q^*(s, a)$. However, for practical problems, it is obviously not feasible to solve the optimal strategy by iterative (5), because in the large state space, the method of solving the Q-value function by the iterative Bellman equation is too expensive [32]. In response to this problem, a linear function approximation is usually used in the RL algorithm to approximate the state-action value function, $Q(s, a|\theta) \approx Q^*(s, a)$. In addition, nonlinear function approximations, such as deep neural networks, can also be used to approximate the value function or strategy.

Gao *et al.* [13] combined the convolutional neural network with the Q learning algorithm in traditional RL, and proposed the deep Q-network (DQN) model. This model is used to process visual perception-based control tasks, and is a ground breaking work in the field of DRL. The training process of DQN is illustrated in Fig. 4.

To alleviate the instability caused by the nonlinear network representation value function, DQN makes three main improvements to the traditional Q learning algorithm.

- DQN uses the empirical playback mechanism during training to process the transferred samples online. The transfer samples obtained by the interaction of the agent with the environment are stored in the playback memory unit at each time step. During training, each small batch of transferred samples is randomly selected and the network parameter ęÈ is updated using a stochastic gradient descent algorithm.
- Not only does DQN use a deep convolutional network to approximate the current value function, but another network is used alone to generate the target Q-value. Specifically, $Q(s, a|\theta)$ represents the output of the current value network, and is used to evaluate the value function of the current state-action pair. $Q(s', a'|\theta_-)$ represents the output of the target value network, and $Y_i = r + \gamma max_{a'} Q(s', a'|\theta_-)$ is generally used to approximate the optimization objective of the value function, i.e., the target Q-value.
- DQN reduces the bonus value and error term to a limited interval, which ensures that the Q-value and the gradient value are within a reasonable range, which improves the stability of the algorithm. Experiments show that DQN exhibits a competitive level comparable to that of human players when solving complex problems such as the Atari 2600 game [25].

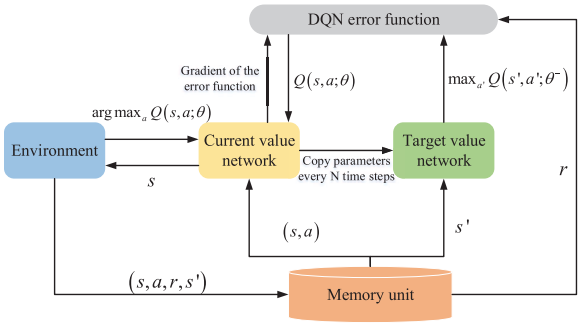

Fig. 4. The deep Q-network (DQN) training steps [10]. The DQN uses the empirical playback mechanism during training to process the transferred samples online.

With the reinforcement learning design, the UAVs interact with the environment without any prior knowledge. The UAVs then learn how to share the information correctly and efficiently by the reward mechanism of reinforcement learning. DQN has a memory library to learn from previous experience. When the DQN is updated, some previous experiences can be randomly selected from the library to learn from. Because the historical data samples are randomly extracted from the library, the correlation between past-experience samples is disrupted and the neural network updates can be more efficient.

D. Long Short-Term Memory (LSTM)

The recursive structure of the long short-term memory (LSTM) network is improved based on the recurrent neural network (RNN). The difference between LSTM and the original RNN is that the LSTM network has four processing modules which interact with each other in a unique way to achieve long-term information processing capability [33]. The LSTM network structure is shown in Fig. 5.

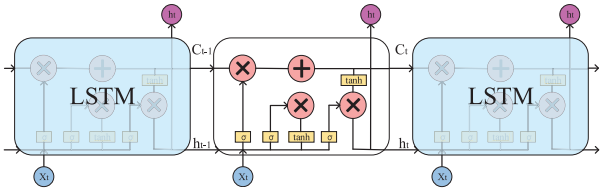

Fig. 5. The LSTM network structure diagram: input value of the current time x_t, output value of the previous time h_t, and unit state of the previous time c_t.

LSTM belongs to the class of time recurrent neural networks (RNN). It is effective for addressing the problems of vanishing gradient and exploding gradient in the process of long-sequence training [22]. There is only one state unit in the hidden layer of the original RNN. This state unit is very sensitive to short-term input, and because of the exponential function, the gradient of the RNN will disappear or explode. LSTM was proposed by Hochreiter and Schmidhuber to solve this problem by adding a state c to the RNN to preserve long-term memory [23].

LSTM uses two gates to control the content of cell state c. One is the forgetting gate, which determines how much cell state of the previous moment c_{t-1} is reserved to the current moment c_t; the other is the input gate, which determines how much input x_t of the network is saved to cell state c_t at the current moment. LSTM uses output gates to control how much cell state c_t is output to the current output value h_t of LSTM.

The forward propagation process of LSTM is as follows. First, the forgotten gate layer determines which information is discarded from the LSTM cell state. The output of the forgotten gate can be expressed as [22]:

$$f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big), \tag{6}$$

where W_f is the weight matrix of the forgetting gate, $[h_{t-1}, x_t]$ is the combination of two vectors into a longer vector, b_f is the bias term of the forgetting gate and σ is the sigmoid function.

Second, the LSTM needs to determine which information is stored in the long-term memory unit of the LSTM, which contains two parts. First, the s-shaped layer of the input gate is used as the information to be updated, and then, the tanh unit will create a new candidate value vector C_t and add it to the unit state. After forgetting, the new state of the unit will be multiplied by the old state C_{t-1} and f_t to obtain information, and C_t can be updated. These steps are expressed as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \tag{7}$$

$$C'_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C), \tag{8}$$

$$C_t = f_t \otimes C_{t-1} + i_t \otimes C'_t, \tag{9}$$

where $\otimes$ represents the element-by-element multiplication of two vectors.

Finally, the output gate updates the state of the LSTM unit. First, the s-shaped layer is used to determine which parts of the current cell state require output; then, tanh processes the cell state to obtain a value between -1 and 1, and multiplies the s-type output to obtain an output value. The arithmetic expression of this processing is as follows:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \tag{10}$$

$$h_t = o_t \otimes \tanh C_t. \tag{11}$$

In this paper, LSTM is mainly used to preserve historical observation data. Specifically, LSTM is used to solve the problem of the weak ability of the circular neural network to process long-term memory information. In the process of reinforcement learning, more historical information can help the UAV to more quickly learn the characteristics of the environment [34]. LSTM networks can also predict environmental feedback from future UAV actions. This can speed up the convergence of the algorithm.

III. Dynamic Management Based on Reinforcement Learning

To improve the convergence speed of the deep reinforcement learning algorithm, a long short-term memory (LSTM) network is introduced to preserve the historical information of action and environmental feedback. By learning the historical information, a more rapid model of the environment can be built to adapt to the highly dynamic characteristics of the UAV cluster environment. In addition, LSTM can also predict the future state of the environment and help DQN learn the optimal strategy more quickly.

A. Algorithm Structure

The UAV dynamic management scheme is subsequently described. The key elements of reinforcement learning in the multi-UAV environment, i.e., agent, action, state, reward function, and strategy, are first defined. The channel allocation strategy is determined by the deep Q-value network, and the agent will gradually provide the optimal strategy through continuous interaction with the environment [35]. In the multi-UAV context, the elements are defined as follows.

Actions (dynamic channel allocation): Suppose there are *N* UAVs. Each individual UAV is defined as an *agent*. There are *N* types of actions for each agent, i.e., sharing information with other $N-1$ UAVs or waiting. The action form of UAV is then defined as:

$$a_t^n \in \{0, 1, \ldots, N-1\}, a_t^n \neq n. \tag{12}$$

Actions (dynamic time slots allocation): Suppose there are *N* UAVs. Each individual UAV is defined as an *agent*. There are two types of actions for each agent, i.e., sharing information with other $N-1$ UAVs or waiting, $a_t = 1$ represent the current request for sharing information. The action form of UAV is then defined as:

$$a_t^n \in \{0, 1\}. \tag{13}$$

Rewards: The reward *R* of the action is defined as the predicted reward function for data transmission. The mean opinion score (MOS) metric is introduced to calculate the reward. The MOS can be calculated as follows:

$$R = MOS = \frac{a_1 + a_2 FR + a_3 ln(SBR)}{1 + a_4 TPER + a_5 (TPER)^3}, \tag{14}$$

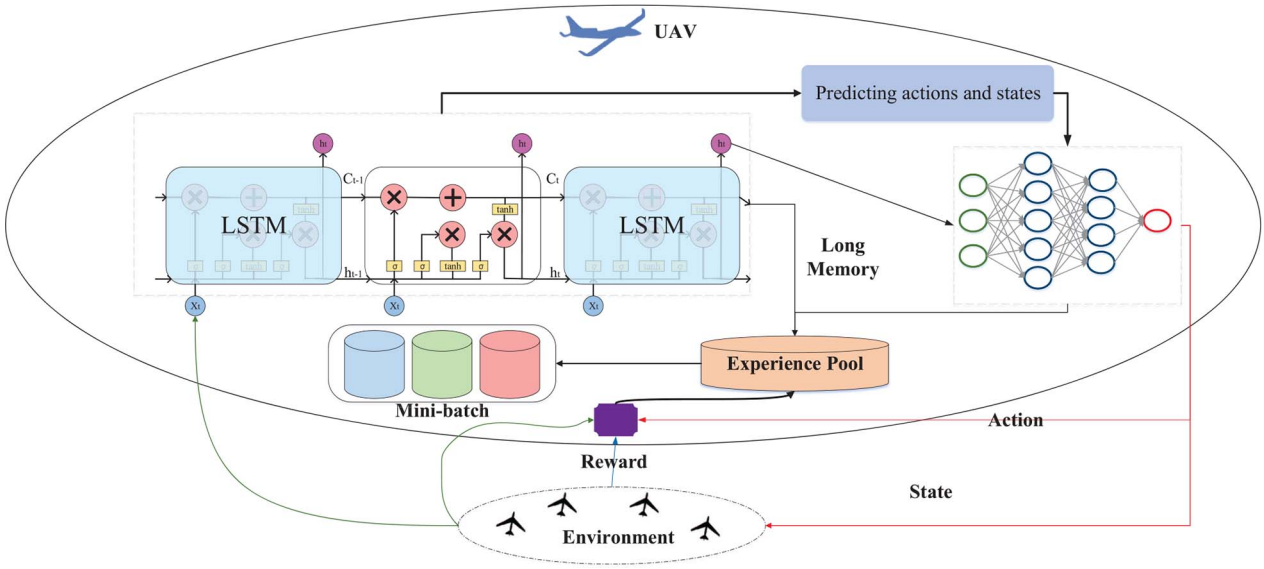

Fig. 6. The algorithmic framework, in which LSTM is integrated with DQN to store the historical action and environment status of the UAV in the LSTM. Environmental feedback is input to the input unit x_t of the LSTM. The output h_t of the LSTM is used as input to the DQN. The long term unite c_t of the LSTM is stored in the experience pool. The environmental feedback includes throughput, collision, packet loss rate and delay.

where FR, SBR and TPER are the frame rate, sending bit rate and total packet error rate, respectively. Additionally, a_i, $i \in 1, 2, 3, 4, 5$ is estimated using the linear regression process. The MOS range is from 1 (lowest) to 5 (highest).

The structure adopted for the algorithm in the present study is an integration of LSTM and DQN, so the iterative updates of the Q-value are similar to that in DQN:

$$Q(s_t^n, a_t^n) \leftarrow Q(s_t^n, a_t^n) + \alpha \left[r_{t+1}^n + \gamma \max_{a_{t+1}^n} Q\left(s_{t+1}^n, a_{t+1}^n\right) - Q(s_t^n, a_t^n) \right], \quad (15)$$

where $\alpha \in [0, 1]$ is the learning rate and $\gamma \in [0, 1]$ is the discounted rate of reward. When the Q-value is updated, the following action principles are chosen:

$$a_{t+1} = \begin{cases} \text{argmax}_a \, Q(s, a), & \text{if } 0 < \varepsilon \leq p_e \\ \text{Choose random action}, & \text{if } p_e < \varepsilon < 1. \end{cases} \quad (16)$$

In (16), $\varepsilon \in (0, 1)$ is a randomly generated number, and p_e is the probability of exploration. Because there are many states in the environment, a decreasing probability of exploration is designed to ensure that there is enough space for exploration at the beginning:

$$p_e = \frac{CTS}{TTS} 0.9, \quad (17)$$

where CTS and TTS are the current time slots and total time step respectively.

Environmental feedback includes throughput, the collision of UAV access, and residual channel capacity. The proposed LSTM+DQN algorithm framework for dynamic channel allocation of the UAV fleet is illustrated in Fig. 6. The feedback of the environment is first input to the input of LSTM. The information is stored and predicted in the network of LSTM. Then the result is input to the neural network for calculation and estimation, and finally the optimal strategy is given. The detailed procedure of the LSTM+DQN algorithm phase for training steps is presented in Algorithm 1.

Algorithm 1 LSTM+DQN Algorithm Phase for Training Steps

Initialize relay memory D to capacity N;
Initialize action-value function Q with random weights θ;
Initialize target action-value function Q_{t+1};
for episode $i = 1, 2, \ldots, M$ **do**
  for time-slot $t = 1, 2, \ldots, T$ **do**
    for each UAV in UAVs **do**
      Initialization of LSTM and DQN networks;
      Generate a random number ε;
      Select actions according to (16);
      Obtain feedback from the environment between each UAV on communication result and reward value r_{t+1};
      The state and action are input into the LSTM network, and the output of LSTM network is used as the input to the DQN network;
      Store transition information in long-term memory unit of LSTM;
      Store transition information in D of DQN;
      The DQN computes the Q-estimation Q_{t+1}, and calculates Q-value according to the actual Q_t;
      Update the Q value according to (15) as:
      $$Q_t \leftarrow Q_t + \alpha \left[r_{t+1} + \gamma \max_{a_{t+1}} Q_{t+1} - Q_t \right]$$
    end for
  end for
end for
return Allocation Strategy

The detailed procedure of the dynamic channel allocation algorithm is presented in Algorithm 2, and the detailed

Algorithm 2 The Dynamic Channel Allocation Algorithm

for episode $i = 1, 2, \ldots, M$ **do**
  for time-slot $t = 1, 2, \ldots, T$ **do**
    for each UAV in UAVs **do**
      Initialization priority for UAV through the task information;
      Initialization actions according to (12);
      Training UAV formations for dynamic channel allocation with Algorithm 1;
    end for
  end for
end for
return Channel Allocation Strategy

Algorithm 3 The Dynamic Time Slots Allocation Algorithm

for episode $i = 1, 2, \ldots, M$ **do**
  for each UAV in UAVs **do**
    Initialization priority for UAV through the task information;
    Initialization actions according to (13);
    Training UAV formations for dynamic time slots allocation with Algorithm 1;
  end for
end for
return Time Slots Allocation Strategy

procedure of the dynamic time slots allocation algorithm is presented in Algorithm 3.

B. Evaluation Factor

If a busy channel is detected as idle, this detection result is called a false alarm. The detection probability is called the false alarm probability, which is an important parameter of the accuracy of spectrum sensing [36]. The effective channel utilization of a UAV can be evaluated using the spectral sensing accuracy and channel hold time (CHT). It is well known that the higher detection probability *pd* corresponds to a lower false alarm probability *pf*. In this way, the spectrum sensing accuracy can be expressed as follows:

$$M_A = P\_d(1 - P\_f). \tag{18}$$

If T represents the total frame length and τ is the channel sensing time, the transmission period is $T - \tau$. It is assumed that the UAV arrival rate λ_{ph} follows a Poisson distribution, and the CHT of duration t has the following probability distribution:

$$f_t = \lambda_{ph} e^{-(\lambda_{ph})t}. \tag{19}$$

From the work by Koushik *et al.* [24], the channel utilization factor can be obtained:

$$CUF = M_A \cdot \frac{t}{T}\left(1 - e^{\left(-\frac{T-t}{t}\right)}\right). \tag{20}$$

The average collision rate (ACR), the MOS, and the throughput (TH) are used to evaluate the performance of the proposed algorithm. The ACR is defined as:

$$ACR = \frac{1}{M}\sum_{i=1}^{M}\frac{n_i}{C}, \tag{21}$$

where M is the total number of training steps, C is the number of channels, and n_i is the number of collision channels during the i-th training step. The MOS is shown in (14).

The normalized throughput (TH) is:

$$(TH_k)_{norm} = \frac{TH_k}{(TH_k)_{ideal}}, \tag{22}$$

where $(TH_k)_{ideal}$ is the ideal throughput calculated via the Shannon capacity theorem.

TABLE I
SIMULATION PARAMETERS

Parameters	Value
Number of UAVs	5
Number of channels	4
Detection probability (P_d)	[0.9, 0.99]
False alarm probability (P_f)	[0.01, 0.1]
Exponential distribution rate (λ_{ph},i={0,1})	[0.02, 1]
Number of training steps	10000, 18000
Discounted rate (γ)	0.01
Learning rate (α)	0.01
Number of priority	5

IV. SIMULATION EVALUATION AND DISCUSSIONS

A. Configuration

In this section, the simulation evaluation of the proposed dynamic management scheme for multi-UAVs systems is presented. The parameters of the experiment were set as shown in Table I.

For the algorithm parameters, 128 hidden layers were chosen for the LSTM network. The length of each stored historical sequence was 5. The number of hidden layers of the DQN neural network was also set to 128. The parameters of the LSTM+DQN, DQN, and Q-learning schemes were set as: a learning rate of $\alpha = 0.01$ a discounted rate of $\gamma = 0.01$. The DQN and Q-learning algorithms were used as baselines to compare with the proposed LSTM+DQN algorithm.

The CUF can be used to represent the results of the spectrum evaluation to select the best channel. According to the IEEE 802.22 recommendation, the probability of correct detection is $P_d = [0.9, 0.99]$ and the possibility of false positives, $P_f = [0.01, 0.1]$. Therefore, the possibility of spectrum sensing accuracy is $P_d(1 - P_f) = [0.81, 0.99]$.

B. Channel Evaluation

The channel environment was evaluated first, including the spectrum sensing accuracy (M_A) and the CUF according to Equation (18) and (20), respectively.

The perceptual accuracy of the spectrum of each UAV was first calculated. As shown in Fig. 7, the perceptual accuracy of each UAV was calculated by Equation (18). The specific parameters are given above.

Fig. 8 presents the channel utilization factor as calculated by Equation (20). Because the scene design used in this study

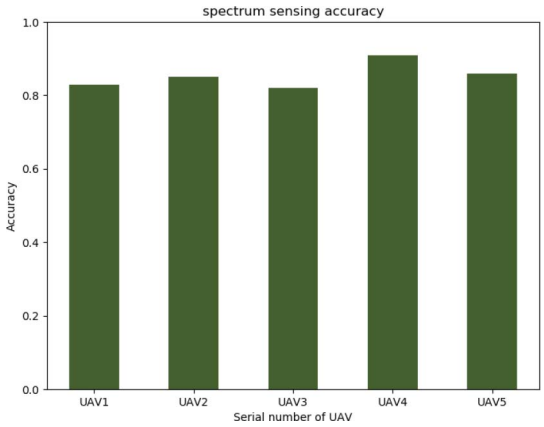

Fig. 7. Spectrum sensing accuracy as calculated by Equation (18).

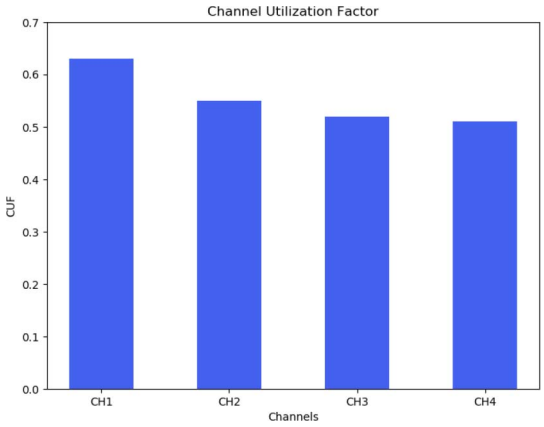

Fig. 8. Channel utilization factor as calculated by Equation (20).

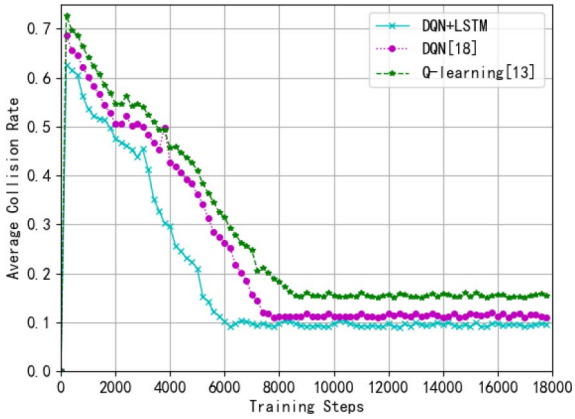

Fig. 9. The average collision rates achieved by Q-learning, DQN [18], and LSTM+DQN (dynamic channel allocation).

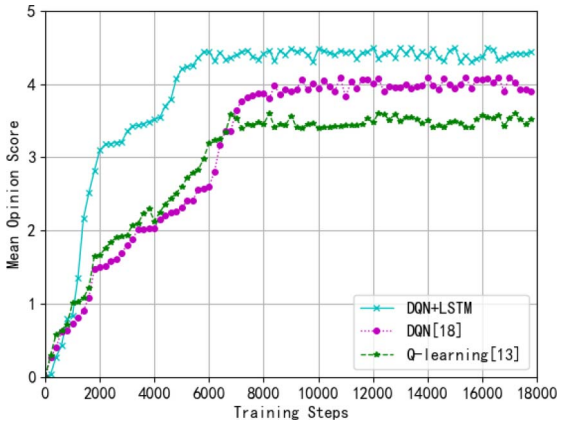

Fig. 10. The mean opinion score achieved by Q-learning, DQN [18], and LSTM+DQN (dynamic channel allocation).

has five UAVs to form a flight formation, up to four channels are used for the transmission of information in each time slot, whether they are dynamically allocated channels for shared information, or dynamically allocated time slots for sharing information. Therefore, the number of channels was set to 4 in the present study, which fully meets the needs of UAV formations. In the process of performing dynamic channel allocation, the channel can be selected, and the channel with high channel quality is selected for communication. When dynamic time slot allocation is performed, each channel will be occupied in a time slot.

C. Dynamic Channel Allocation

To evaluate the performance of the three algorithms, 200 training samples were collected during each iteration to calculate the average collision rate (ACR), the mean opinion score (MOS), and the throughput (TH). The three evaluation index curves of the three algorithms (Q-learning, the deep Q-network (DQN) [18], and the proposed LSTM+DQN) are presented in Figs. 9–11.

As can be seen from Fig. 9, the ACR of the three algorithms did not differ much at the beginning of the experiment. This is because the feedback that the agent obtained from the environment was not sufficient when the algorithm interacted with the environment. As the number of training increased, UAVs accumulated increasingly more information from the environment. It can be seen from the figure that the LSTM+DQN algorithm proposed in this paper converged after 6000 time slots, which is the fastest convergence rate among the three schemes. The other two algorithms began to converge after 8000 time slots and 9000 time slots, respectively. Compared with the traditional algorithm, the convergence speed of the algorithm was nearly 2000 steps. This is because the LSTM network retained more historical information, enabling the DQN neural network to better predict the optimal channel access strategy. After reaching convergence, the average collision rate of the method fluctuated around 10%, while the collision rates of the other two algorithms were 12% and 16%, respectively. The reason for the collision probability fluctuation is that there is a possibility of $1 - p_e$, i.e., reinforcement learning will select random actions to prevent local optimality in the process.

Fig. 10 shows the mission management system for the UAV. The MOS value indicates whether the allocation strategy obtained is the optimal strategy. According to the experimental results of Fig. 10, the LSTM+DQN algorithm proposed in this paper obtained the highest MOS; after convergence, it was 13% higher than that of DQN algorithm. In addition, the

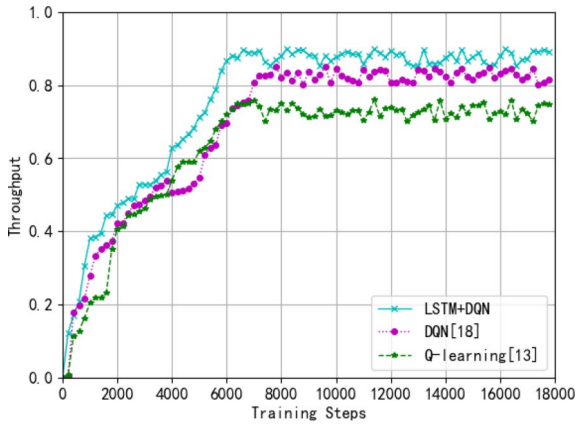

Fig. 11. The throughput achieved by Q-learning, DQN [18], and LSTM+DQN (dynamic channel allocation).

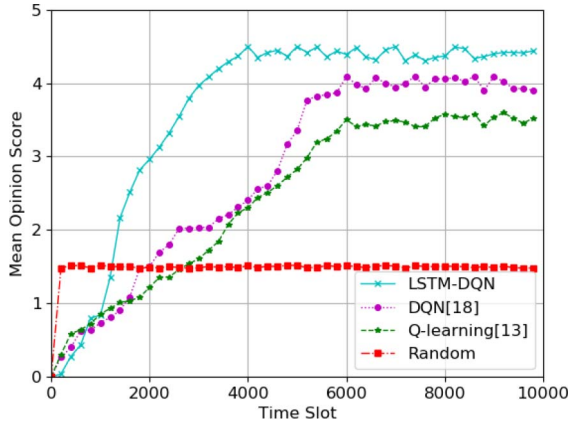

Fig. 12. The mean opinion score achieved by Q-learning, DQN [18], and LSTM+DQN (dynamic time-slot allocation).

maximum likelihood of the Q-learning method was about 20% lower than the other two algorithms. This is because the first two algorithms have a memory library for learning historical information, which helps to find the optimal strategy. Because LSTM can store historical information for a long time, its MOS convergence speed was 2000 steps faster than that of the other two algorithms.

The TH in these three schemes is shown in Fig. 11. In this study, both conflict and non-access were considered communication failures. It can be seen from the curve that the TH of the proposed algorithm was 5% higher than that of DQN, and 10% higher than that of Q-learning. Although the improvement in this scenario is modest, it demonstrates the advantages of the proposed algorithm.

The improved algorithm uses LSTM to process the historical information, which not only improves the learning ability of the network to the historical information, but also can make a certain prediction through the historical information, which greatly improves the convergence speed of the network. Based on the preceding results and discussions, it can be concluded that the proposed LSTM+DQN algorithm achieves a faster convergence speed and a superior channel allocation strategy compared to the two baseline schemes.

D. Dynamic Time Slot Allocation

The difference between dynamic time slot allocation and dynamic channel allocation is that the channel is used without collision in each time slot, i.e., only one UAV is transmitting information, and other UAVs are in the state of reception, Therefore, only MOS and TH, not ACR, were considered in this study, as shown in Figs. 12 and 13. Additionally, the prioritized UAV formation will use the M/G/1 queuing model to generate queuing during the time period in which the waiting time slot is available, so this UAV will cause packet loss due to a long queue time. In the UAV formation task, the information obtained by the main work module is necessary, some auxiliary information is allowed to be lost, thus, task delay for each UAV was evaluated, and the results are presented in Fig. 14.

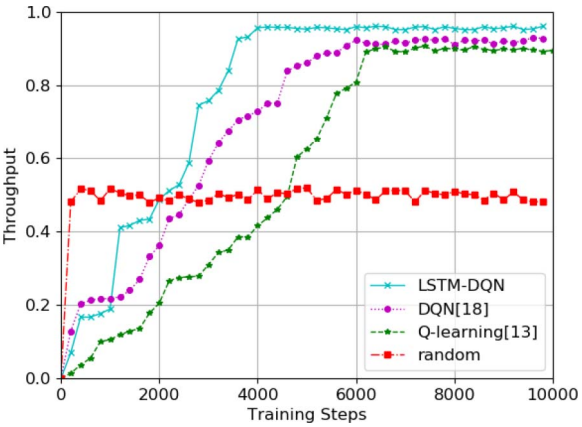

Fig. 13. The throughput achieved by Q-learning, DQN [18], and LSTM+DQN (dynamic time-slot allocation).

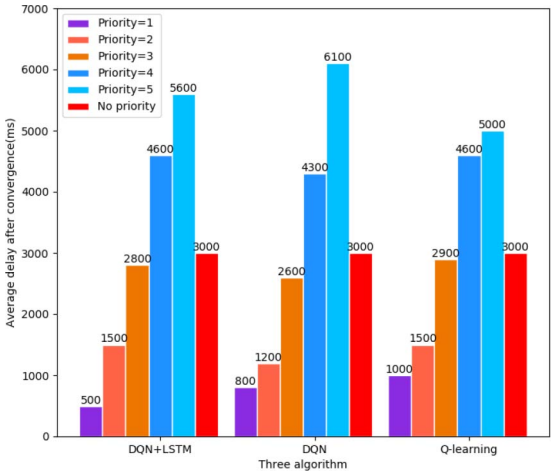

Fig. 14. The delay achieved by LSTM+DQN. Comparison of priority access methods and non-priority random access methods.

As can be seen from Fig. 12, the convergence rate of the average bonus value of the dynamic time slot allocation was faster than that of dynamic channel allocation, and convergence began about 2000 steps in advance. This is because there is no need to consider collisions between channels in the process of dynamically allocating time slots, as all

channels in each time slot are used. What needs to be assigned is only which UAV is used to send information in this time slot, so the remaining UAVs are all adjusted to accept the status of the information. It can be seen from the figure that the proposed algorithm had a faster convergence speed, which means that the training time was shorter, and the optimal strategy could be given faster; it was 2000 training steps faster than the other two algorithms. The value of MOS represents the satisfaction with the strategy. The highest MOS value is 5. When the proposed algorithm converges, the value of MOS can be stabilized at about 4.4, which is 10% higher than that of the DQN algorithm 4.0 and 26% higher than that of the DQN algorithm 3.5.

The MOS value that the system can achieve in the case of randomly assigned time slots was also analyzed. It can be seen from the curve that the random allocation is very stable, but its MOS value was only about 1.5, which is far less than half that of the proposed algorithm.

Fig. 13 presents the throughput of the dynamic time slot allocation system. The throughput of dynamic time slot allocation is also different from that of dynamic channel allocation. Because each time slot channel is occupied, it is mainly determined whether each time slot is fully utilized. The proposed algorithm must ensure that the time slots are used continuously without being wasted. Because a priority approach was introduced, high-priority UAVs cannot be interrupted by low-priority UAVs when they are internships, which causes low-priority UAVs to enter a wait state if the high-priority UAV exits using the time slot, and the low priority UAV may have packet loss, which causes the time slot to be idle and wasted. This kind of waste is inevitable, so the proposed algorithm must solve this issue. As long as the low-priority UAV detects that the time slot is idle, it will occupy the information sharing.

From Fig. 13, it is evident that the proposed algorithm began to converge at 4000 training steps, and the TH at convergence was about 0.98. The other two algorithms reached convergence in 6000 steps, and their throughput rates were about 0.92 and 0.9, respectively. The throughput rate of the proposed algorithm was respectively 6% and 8% higher than those of the other two algorithms, which appears to be not very obvious. This is because the dynamic time slot allocation process is relatively simpler than the dynamic channel allocation process, so all three algorithms can achieve better results.

Additionally, experiments of random time slot allocation were also conducted. From the curve, it can be concluded that random allocation has a very stable effect, but the TH always stays at 50%, this means that half of the time slots have no information. Transmission tasks, which greatly waste the communication resources, are not conducive to the timeliness of the information sharing of UAV formations, and may have a relatively large impact on the success of the final mission of the UAVs.

In this paper, the UAV service time was set to obey an exponential distribution, and the number of channels was set to 4. The specific parameter settings are shown in Table II.

TABLE II
PRIORITY PARAMETERS

Priority	Number of UAV	Transmission Rate(MHz)	Delay(ms)
1	1	3.6	30
2	1	2.3	60
3	1	1.5	100
4	1	1.0	1000
5	2	<1.0	No

When evaluating the delay, only that of the LSTM+DQN algorithm was measured. Because the delay measures the role of the priority mechanism, different priorities represent different important tasks, i.e., modules with different functions carried on the UAV. The higher the priority, the more important the tasks undertaken in the formation. Therefore, in principle, the information collected by these modules cannot be missing. On the contrary, the information collected by the lowest priority module is used as the entire formation task. Auxiliary information can generate packet loss, which will not have a serious impact on the success of the task.

The delay in this scheme is shown in Fig. 14. It can be seen from the figure that after adding priority, the three algorithms can improve the service delay of high priority UAVs. For the highest priority UAVs, the three algorithms reduce the delay by 83%, 73% and 67% respectively. Among them, the improved dqn + LSTM algorithm has better results in high priority improvement. Although the Q-learning algorithm has been improved, it can be found that it does not obviously reflect the level restrictions brought by the priority, while the dqn algorithm has very obvious level restrictions. It can be seen that the lowest priority UAV delay has reached 6100ms, which can almost be identified as the basic UAV There is little access to transmit information on.

V. CONCLUSION

This paper examined a system model for the dynamic management of UAV flight formations, and studied how UAV formations share information in the absence of missions. Two models were analyzed, one was the dynamic allocate channels for sharing purposes, and the other was the exchange of information between UAVs by dynamically allocating time slots. The M/G/1 queuing model was designed to measure the UAV packet loss and latency, and the MOS was used to design reward functions in reinforcement learning. Three kinds of reinforcement learning algorithms were studied, namely Q-learning, DQN and the proposed LSTM+DQN. Through the simulation results, it was concluded that the proposed algorithm has fast convergence ability and excellent performance in both the dynamic channel allocation model and the dynamic time slot allocation model. In the dynamic channel allocation model, the convergence speed was about 2000 steps, and the ACR was 2% and 6% lower than that of other two algorithms. The value of MOS was 13% and 20% higher than that of other two algorithms. The value of TH was also better than that of other two algorithms by 5% and 10%, respectively. In the dynamic time slot allocation model, the MOS value of the proposed algorithm was 10% and 26% higher than that of the

other two algorithms, and the TH value was 6% and 8% higher, respectively. Based on the reinforcement learning algorithm, a priority mechanism was introduced to ensure that important information was not lost. From the simulation results, it was found that the average latency of the two UAVs with the highest priority was 83% lower than that of the non-priority model.

In future work, the average delays of three low-priority UAVs were found to be higher than that of the non-priority model, the authors will continue to ensure that other priority UAVs occupy resources reasonably under the premise of ensuring the priority while ensuring the high priority UAVs fully occupy resources.

REFERENCES

[1] R. Wang and L. Jinkun, "Adaptive formation control of quadrotor unmanned aerial vehicles with bounded control thrust," *Chin. J. Aeronautics*, vol. 30, no. 2, pp. 807–817, 2017.

[2] Z. Wang, L. Liu, and T. Long, "Multi-UAV reconnaissance task allocation for heterogeneous targets using an opposition-based genetic algorithm with double-chromosome encoding," *Chin. J. Aeronautics*, vol. 31, no. 2, pp. 339–350, 2018.

[3] Z. Xiao and D. Ling, "Fast deployment of UAV networks for optimal wireless coverage," *IEEE Trans. Mobile Comput.*, vol. 18, no. 3, pp. 588–601, Mar. 2019.

[4] G. Yang, Z. Yi, W. Hao, and C. Xin, "Study on an average reward reinforcement learning algorithm," *Chin. J. Comput.*, vol. 30, no. 8, pp. 1372–1378, 2011.

[5] F. Ming, L. Quan, W. Hui, X. Fei, Y. Jun, and L. Jiao, "A novel off policy Q(λ) algorithm based on linear function approximation," *Chin. J. Comput.*, vol. 37, no. 3, pp. 677–686, 2014.

[6] J. Kober and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 97, no. 11, pp. 9–67, 2014.

[7] A. T. Hafez, M. Iskandarani, S. N. Givigi, S. Yousefi, and A. Beaulieu, "UAVs in formation and dynamic encirclement via model predictive control," *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 1241–1246, 2014.

[8] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, 2nd Quart., 2016.

[9] E. Ipek, O. Mutlu, and J. F. Martinez, "Self-optimizing memory controllers: A reinforcement learning approach," *ACM SIGARCH Comput. Architect. News*, vol. 36, no. 3, pp. 39–50, 2008.

[10] R. Sutton and A. Barto, "Reinforcement learning: An introduction," *CSEE J. Power Energy Syst.*, vol. 4, no. 3, pp. 362–370, 2018.

[11] J. Ye, C. Zhang, H. Lei, G. Pan, and Z. Ding, "Secure UAV-to-UAV systems with spatially random UAVs," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 564–567, Apr. 2019.

[12] S. Zhang, H. Zhang, B. Di, and L. Song, "Cellular UAV-to-X communications: Design and optimization for multi-UAV networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 817–825, Feb. 2019.

[13] Z. Gao, B. Wen, L. Huang, C. Chen, and Z. Su, "*Q*-learning-based power control for LTE enterprise femtocell networks," *IEEE Syst. J.*, vol. 11, no. 4, pp. 2699–2707, Dec. 2017.

[14] J. Ni, M. Liu, L. Ren, and S. X. Yang, "A multiagent *Q*-learning-based optimal allocation approach for urban water resource management system," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 1, pp. 204–214, Jan. 2014.

[15] H.-H. Chang, H. Song, Y. Yi, J. Zhang, H. He, and L. Liu, "Distributive dynamic spectrum access through deep reinforcement learning: A reservoir computing-based approach," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1938–1948, Apr. 2019.

[16] A. R. Syed, K.-L. A. Yau, J. Qadir, H. Mohamad, N. Ramli, and S. L. Keoh, "Route selection for multi-hop cognitive radio networks using reinforcement learning: An experimental study," *IEEE Access*, vol. 4, pp. 6304–6324, 2016.

[17] V. Raj, I. Dias, T. Tholeti, and S. Kalyani, "Spectrum access in cognitive radio using a two-stage reinforcement learning approach," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 20–34, Jun. 2018.

[18] N. Oshri and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 310–323, Jan. 2019.

[19] J. Lundén, S. R. Kulkarni, V. Koivunen, and H. V. Poor, "Multiagent reinforcement learning based spectrum sensing policies for cognitive radio networks," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 5, pp. 858–868, Oct. 2013.

[20] D. Silver *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[21] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," *Comput. Sci.*, vol. 8, no. 6, pp. 187–189, 2016.

[22] A. Diro and N. Chilamkurti, "Leveraging LSTM networks for attack detection in fog-to-things communications," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 124–130, Sep. 2018.

[23] J. Zhao, Y. Gao, and Z. Bai, "Traffic speed prediction under non-recurrent congestion: Based on LSTM method and BeiDou navigation satellite system data," *IEEE Intell. Transp. Syst. Mag.*, vol. 11, no. 2, pp. 70–81, Mar. 2019.

[24] A. M. Koushik, F. Hu, and S. Kumar, "Intelligent spectrum management based on transfer actor-critic learning for rateless transmissions in cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 5, pp. 1204–1215, May 2018.

[25] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[26] J. Wen and X.-W. Chang, "On the KZ reduction," *IEEE Trans. Inf. Theory*, vol. 65, no. 1, pp. 1921–1935, Mar. 2019.

[27] J. Wen, Z. Zhou, Z. Liu, M. Lai, and X. Tang, "Sharp sufficient conditions for stable recovery of block sparse signals by block orthogonal matching pursuit," *Appl. Comput. Harmonic Anal.*, vol. 47, no. 1, pp. 948–974, 2019.

[28] Y. Lin, X. Zhu, and Z. Zheng, "The individual identification method of wireless device based on dimensionality reduction and machine learning," *J. Supercomput.*, vol. 75, no. 6, pp. 3010–3027, 2017, doi: 10.1007/s11227-017-2216-2.

[29] Y. Tu, Y. Lin, and J.-K. Wang, "Semi-supervised learning with generative adversarial networks on digital signal modulation classification," *Comput. Mater. Continua*, vol. 55, no. 2, pp. 243–254, 2018.

[30] L. Yun, C. Wang, J. Wang, and Z. Dou, "A novel dynamic spectrum access framework based on reinforcement learning for cognitive radio sensor networks," *Sensors*, vol. 16, no. 10, pp. 1–22, 2016, doi: 10.3390/s16101675.

[31] Z. Dou, G. Si, Y. Lin, and M. Wang, "An adaptive resource allocation model with anti-jamming in IoT network," *IEEE Access*, vol. 7, pp. 93250–93258, 2019.

[32] G. Ding, Q. Wu, L. Zhang, Y. Lin, T. A. Tsiftsis, and Y.-D. Yao, "An amateur drone surveillance system based on the cognitive Internet of Things," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 29–35, Jan. 2018.

[33] Z. Xue, J. Wang, G. Ding, Q. Wu, Y. Lin, and T. A. Tsiftsis, "Device-to-device communications underlying UAV-supported social networking," *IEEE Access*, vol. 6, pp. 34488–34502, 2018.

[34] S. K. Haider, A. Jiang, M. A. Jamshed, H. Pervaiz, and S. Mumtaz, "Performance enhancement in P300 ERP single trial by machine learning adaptive denoising mechanism," *IEEE Netw. Lett.*, vol. 1, no. 1, pp. 26–29, Mar. 2019.

[35] Z. Zheng, T. Wang, J. Wen, S. Mumtaz, A. K. Bashir, and S. H. Chauhdary, "Differentially private high-dimensional data publication in Internet of Things," *IEEE Internet Things J.*, early access, doi: 10.1109/JIOT.2019.2955503.

[36] B. Zheng *et al.*, "Design of multi-carrier LBT for LAA&WiFi coexistence in unlicensed spectrum," *IEEE Netw.*, vol. 34, no. 1, pp. 76–83, Jan./Feb. 2020.

[37] Y. Jiang *et al.*, "Joint power and bandwidth allocation for energy-efficient heterogeneous cellular networks," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6168–6178, Sep. 2019.

[38] Y. Zou, T. Wu, M. Sun, J. Zhu, M. Qian, and C. Liu, "Secrecy outage analysis of non-orthogonal spectrum sharing for heterogeneous cellular networks," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6626–6640, Sep. 2019.

[39] Y. Jiang, Y. Zou, J. Ouyang, and J. Zhu, "Secrecy energy efficiency optimization for artificial noise aided physical-layer security in OFDM-based cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 11858–11872, Dec. 2018.

[40] Y. Zou, "Intelligent interference exploitation for heterogeneous cellular networks against eavesdropping," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 7, pp. 1453–1464, Jul. 2018.

[41] P. Yan, Y. Zou, and J. Zhu, "Energy-aware multiuser scheduling for physical-layer security in energy-harvesting underlay cognitive radio systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2084–2096, Mar. 2018.

Yun Lin (Member, IEEE) received the B.S. degree from Dalian Maritime University, in 2003, the M.S. degree from the Harbin Institute of Technology, in 2005, and the Ph.D. degree from Harbin Engineering University, China, in 2010, where he is currently a Full Professor with the College of Information and Communication Engineering. He was a Visiting Scholar with the Broadband Mobile Wireless Communications Research Group, Department of Electrical Engineering, Wright University, Dayton, OH, USA. He has published more than 150 international peer-reviewed journal/conference papers, such as the IEEE INTERNET OF THINGS JOURNAL, the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING, the IEEE TRANSACTIONS ON RELIABILITY, IEEE ACCESS, INFOCOM, GLOBECOM, ICC, VTC, and ICNC. His current research interests include machine learning and data analytics over wireless networks, signal processing and analysis, cognitive radio and software defined radio, artificial intelligence, and pattern recognition. He is serving on the editorial boards of several journals, including the *KSII Transactions on Internet and Information Systems* and the *International Journal of Performability Engineering*, and the Guest Editor for prestigious journals, such as the IEEE TRANSACTIONS ON RELIABILITY, IEEE ACCESS, and *ACM Springer Mobile Networks and Applications*. In addition, he served as the General Chair of ADHIP 2020, the TPC Chair of MOBIMEDIA 2020, ICEICT 2019 and ADHIP 2017, and the TPC Member of many IEEE international conferences, including GLOBECOM, ICC, ICNC, VTC, and SPAWC.

Meiyu Wang received the B.S. degree in electronic information engineering from Northeast Petroleum University in 2015. She is currently pursuing the Ph.D. degree in communication engineering with Harbin Engineering University, Harbin, China. Her research interests include transfer learning, wireless communications, and spectrum prediction.

Xianglong Zhou received the B.S. degree in measurement and control technology and instrumentation (signal processing and instrumentation) from Jilin University in 2017. He is currently pursuing the M.S. degree in communication engineering with Harbin Engineering University, Harbin, China. His research interests include reinforcement learning, wireless communications, and dynamic spectrum allocation.

Guoru Ding (Senior Member, IEEE) received the B.S. (Hons.) degree in electrical engineering from Xidian University, Xi'an, China, in 2008, and the Ph.D. (Hons.) degree in communications and information systems from the College of Communications Engineering, Nanjing, China, in 2014.

He is currently an Associate Professor with the College of Communications Engineering, Nanjing, China. From 2015 to 2018, he was a Postdoctoral Research Associate with the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China. His research interests include cognitive radio networks, massive MIMO, machine learning, and data analytics over wireless networks. He has received the Excellent Doctoral Thesis Award of the China Institute of Communications in 2016, the Alexander von Humboldt Fellowship in 2017, the Excellent Young Scientist of Wuwenjun Artificial Intelligence in 2018, the 14th IEEE COMSOC Aisa-Pacific Outstanding Young Researcher Award in 2019, the Natural Science Foundation for Distinguished Young Scholars of Jiangsu Province, China, and six best paper awards from international conferences such as the IEEE VTC-FALL 2014. He has served as a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (special issue on spectrum sharing and aggregation in future wireless networks). He is currently an Associate Editor of the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING and a Technical Editor of the IEEE 1900.6 Standard Association Working Group.

Shiwen Mao (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from Polytechnic University, Brooklyn, NY, USA, in 2004. He is the Samuel Ginn Professor and the Director of the Wireless Engineering Research and Education Center, Auburn University, Auburn, AL, USA. His research interests include wireless networks, multimedia communications, and smart grid. He is a recipient of the Auburn University Creative Research and Scholarship Award in 2018, the NSF CAREER Award in 2010, several conference best paper awards, and the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems.