An Actor-Critic Deep Reinforcement Learning Approach for Transmission Scheduling in Cognitive Internet of Things Systems

Article *in* IEEE Systems Journal · March 2019

DOI: 10.1109/JSYST.2019.2891520

CITATIONS
16

READS
412

2 authors, including:

Helin Yang
Nanyang Technological University
64 PUBLICATIONS **902** CITATIONS

SEE PROFILE

An Actor-Critic Deep Reinforcement Learning Approach for Transmission Scheduling in Cognitive Internet of Things Systems

Helin Yang, *Student Member, IEEE*, and Xianzhong Xie, *Member, IEEE*

***Abstract*—The cognitive Internet of Things (CIoT) has attracted much interest recently in wireless networks due to its wide applications in smart cities, intelligent transportation systems, and smart metering networks. However, how to smartly schedule the packet transmission in CIoT systems is still a key challenge, that is, how to design a smart agent to realize the intelligent decision making and effective interoperability. In this paper, we model the system state transformation as a Markov decision process, and an actor-critic deep reinforcement learning algorithm based on a fuzzy normalized radial basis function neural network (called AC-FNRBF) is proposed to efficiently solve the intelligent transmission scheduling problem in CIoT systems under high-dimensional variables. The proposed AC-FNRBF algorithm can better approximate both the action function of the actor and the state–action value function of the critic without requiring the system prior knowledge, and a new reward function is established to maximize the system benefit, which jointly takes the transmission packet rate, the system throughput, the power consumption, and the transmission delay into account. Moreover, the AC-FNRBF has the ability to adjust its learning structure and parameters in dynamic environments. Simulation results verify that the proposed algorithm achieves higher transmission packet rate and system throughput with lower power consumption and transmission delay, compared with other existing reinforcement learning algorithms.**

***Index Terms*—Actor-critic (AC), adaptive fuzzy neural network, cognitive Internet of Things (CIoT), deep reinforcement learning (DRL), transmission scheduling.**

I. Introduction

WITH a large number of application services of devices (e.g., mobile phones, vehicles, monitors, sensors, and industrial machines) in wireless communication networks [1], the Internet of Things (IoT) has been emerging as a promising vision for next-generation wireless networks through realizing industrial and factory automation [2]. With the help of the IoT, all these smart devices can be intelligently interconnected to the Internet, as well as exchange vast information with each other. However, the deployments of the IoT are still not smart enough to complete massive data analysis and cognitive decision making, so the IoT is just like an awkward stegosaurus who needs the human beings' cognition intervention [3], [4]. In order to solve this problem, a powerful paradigm called cognitive Internet of Things (CIoT) was first proposed to equip the IoT with an intelligent brain to make decisions by itself [3], [4], which makes the IoT with high-level intelligence.

In the CIoT network, the IoT has the cognitive ability to smartly manage its interoperation and make intelligent decisions independently among smart devices with minimal human interaction [5]. However, one key challenge in the CIoT system is how to design an effective learning agent to intelligently make decisions (such as resource allocation, spectrum access, energy management, and transmission scheduling) for wireless networks under different devices, ambience, and social behaviors. Recently, many researchers have proposed their novel approaches to solve the spectrum access and transmission scheduling problems in CIoT systems [6]–[11]. The literature [6]–[8] investigated the spectrum access problem under different kinds of constraints in the CIoT system, such as enormous connectivity demands [6], minimum reactive jamming attacks and transmission delay [7], and system fairness [8]. Moreover, considering that machine-type devices and human-type devices exist in CIoT systems, a behavioral framework of cognitive hierarchy theory was proposed to solve the distributed resource allocation problem under IoT devices' quality-of-service requirements [9]. In addition, the authors of [10] and [11] proposed two cognitive medium access control (MAC) protocols, to efficiently provide communication services and data transmission with consideration for energy efficiency, reliability, and Internet connectivity in CIoT systems. Even if the above works [6]–[11] improve the system performance, they are still not smart enough to search the optimal policy. In addition, most of the popular optimization technologies [6]–[11] are not suitable for large-scale IoT systems, due to the dynamic characteristics of mobile networks and the diverse service requirements of IoV devices.

Model-free reinforcement learning (RL) is a powerful program for policy selection and decision making in wireless networks [12]–[16], where an agent aims to intelligently learn/search the optimal policy to maximize the system benefit or minimize the system cost. This learning tool has already been widely adopted in wireless communication systems for packet

Manuscript received August 6, 2018; revised October 29, 2018 and December 22, 2018; accepted January 5, 2019. This work was supported in part by the National Nature Science Foundation of China under Grant 61502067 and in part by the Key Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-K201800603. *(Corresponding author: Xianzhong Xie.)*

H. Yang is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: hyang013@e.ntu.edu.sg).

X. Xie is with the Chongqing Key Laboratory of Computer Network and Communication Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: xiexzh@cqupt.edu.cn).

Digital Object Identifier 10.1109/JSYST.2019.2891520

transmission, resource management, and dynamic multichannel access [12]–[16], which achieves the near optimal performance. For example, in [17], a novel radio resource allocation scheme based on RL and the deep neural network was proposed to improve the cognitive satellite communication performance under the continuous multidimensional action–state space.

Recently, many studies have adopted the RL program [Q-leaning, actor-critic (AC), and deep reinforcement learning (DRL)] to address the smart resource allocation problems in CIoT systems [4], [18]–[26]. In RL-based-CIoT systems, a smart agent makes a close interaction with the environment and updates its action based on the current state. After that, the agent receives a reward to evaluate whether the learning policy is good or not; then, it chooses better actions to maximize the reward. Thus, the intelligent decision-making strategy can be achieved during this learning process. Ferreira *et al.* [18] presented a novel green resource allocation approach based on DRL to maximize the transmission packet rate (TPR) with consideration for the satisfaction of quality of experience among devices in content-centric IoT systems. In [19], the authors applied the R-learning approach to control power consumption levels in the CIoT system, where each device can teach other devices to decrease the computational complexity and enhance the learning process. The Q-learning tool is usually adopted in CIoT/IoT systems to make intelligent decisions [20]–[22]. Q-learning-scheme-based cognitive routing was investigated for the cognitive Internet of vehicles (CIoV) in dynamic wireless vehicular environments [20], where the CIoV system exploits learning strategies to efficiently schedule the transmission dataset. Wang *et al.* [21] developed a new deep Q-learning-based transmission scheduling scheme to maximize the system throughput in CIoT systems, and Zhu *et al.* [22] proposed a distributed multiagent Q-learning framework for intelligent traffic scheduling in a big IoT vehicular network. The authors of [23] developed a practical RL-framework-based duty cycle control to improve the energy efficiency and transmission reliability in machine-to-machine IoT systems. Moreover, in [5] and [24]–[26], the authors provided an overview of the existing decision-theoretic models in CIoT systems, such as hybrid or multiple RL models and different kinds of game theory models based on RL and genetic algorithm RL models; these RL models can smartly make decisions of transmission scheduling, resource allocation, and autonomous control.

The above-cited works [20]–[22] based on Q-learning or deep Q-learning approaches achieve a considerable transmission scheduling performance by exploiting optimal learning strategies, but the Q-learning approach [20] has low learning rate when the state space is high dimensional, and the deep Q-learning [21] is not always capable for CIoT systems, due to dynamic mobile environments and continuous variables.

To address the above-mentioned issues, in this paper, a model-free AC RL approach is exploited to search the intelligent transmission scheduling strategy in CIoT systems. The AC RL approach [14], [15], [27]–[30] has the ability to address the problem with continuous state and action variables, where the actor can exploit the actions, and the critic is capable of estimating the value function. In practical CIoT systems, a significant number of IoT devices generate large data sets, leading to the high dimensional system state, which makes system states and transitions hard to determine through learning. Hence, an AC DRL approach based on a fuzzy normalized radial basis function neural network (called AC-FNRBF) is proposed to approximate both the action function of the actor and the state–action value function of the critic. The main contributions of this paper can be summarized as follows.

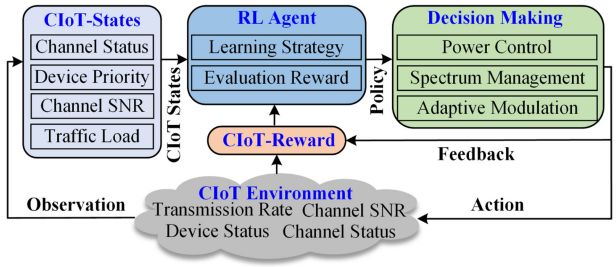

Fig. 1. Intelligent decision-making-model-based CIoT system.

1) The model-free AC DRL approach is first exploited to solve the intelligent transmission scheduling problem in CIoT systems with continuous state and action spaces, and a new reward function is presented to maximize the system reward.
2) In order to approximate both the action function of the actor and the state–action value function of the critic under the high dimensionality of system state spaces and the large number of transmission packets, an AC-DRL algorithm based on the FNRBF is proposed in CIoT systems, called AC-FNRBF, which can solve the transmission scheduling problem with the lack of precise mathematical models. The proposed AC-FNRBF algorithm learns the state transition without the system's prior knowledge, so it avoids large computing and storage in the learning process.
3) Considering the dynamic changes of system states and colorful characteristics of social behaviors in CIoT systems, a self-organizing contracture of the AC-FNRBF algorithm is established to dynamically adjust its structure and parameters during the learning process.

The rest of this paper is organized as follows. In Section II, we present the system model. In Section III, we formulate the transmission scheduling problem. The solution for the optimization problem is presented in Section IV. The simulation results and analysis are discussed in Section V. Finally, Section VI concludes this paper.

II. System Model and Preliminaries

In this section, we will investigate the decision-making framework of the CIoT system. After that, we show several models that we will use for the system performance evaluation and the intelligent decision making in the CIoT system.

The decision-making framework for the wireless CIoT system is shown in Fig. 1. The goal of the CIoT system is to intelligently provide services/applications with the minimum human inter-

face. First, the system observes the environment information [e.g., devices priority, traffic load of channels, channel signal-to-noise ratio (SNR), and channel status (busy or idle)] in the physical environment and sends these important observation results to the upper layer (e.g., MAC layer). These observed pieces of information are called CIoT states (e.g., channel status, channel SNR, device status, and traffic packet load); these states will be described in the following models. The IoT device uses this system state information from the environment to make decision for intelligent transmission scheduling in the RL framework.

After that, the service performance evaluation process is responsible for service provisioning based on the requirements of the social networks. This process builds the evaluation framework to record and evaluate the policy by collecting and analyzing the feedback from social networks. The smart action includes the transmission power control, spectrum allocation and access, spectrum handoff, and adaptive modulation. Thus, in this cognitive framework, the optimal policy and autonomous scheduling can be carried out to support the social network services' requirements and provide considerable actions to the physical environment.

The cognitive decision making can be defined as the process of searching an optimal policy/action to maximize the system reward with considering the transmit packet rate (TPR), system throughput, power consumption, and transmission delay, which we will discuss in the next section.

A. Channel State Model

The packet arrival rate in the CIoT system can be modeled as a Poisson distribution process. The channel occupancy state of N frequency-domain channels follows a discrete-time Markov process. Let $H_n(t)$ denote the channel state of the nth channel over one point-to-point link at the tth time slot. $H_n(t)$ is a binary variable, $H_n(t) \in \{0, 1\}$, representing whether the nth channel is busy or not. If $H_n(t) = 1$, the channel is occupied by one transmission packet; otherwise, the channel is idle. The device can access the nth channel to send the data packet if the channel is idle. In contrast, the device will wait in the queuing list or handoff to other available channels if the channel is busy currently. Note that the channel state can be sensed or detected by devices in the CIoT physical environment by using the cognitive radio (CR) technique [4] and then send the observation results to the upper layer.

Let SNR_n^i denote the instantaneous SNR of the ith channel state on channel n. The corresponding SNR value of each channel varies in different channel states. In other words, it is determined by the received signal power, the channel gain, and the background noise, which is modeled by the frequency-selective fading model in this paper [21].

B. Power Consumption Model

In each time slot, each device has two power consumption status $x \in \{\mathrm{ON}, \mathrm{OFF}\}$, where $x = \mathrm{ON}$ indicates that the power consumption level is active and the device sends data packets, and $x = \mathrm{OFF}$ means that the power consumption status is sleep. When $x = \mathrm{OFF}$, the devices will turn to be at a low-power level to reduce the power consumption under this situation and wait to be scheduled to other channels or wait in the queuing list to access this channel. Consequently, the power consumption level on the nth channel is modeled as

$$P_n = \begin{cases} P_{\mathrm{cir}} + P_n^{\mathrm{tx}}, & \text{if } x = \mathrm{ON}, \text{sendpackets} \\ P_{\mathrm{cir}}, & \text{if } x = \mathrm{OFF}, \text{sleep} \end{cases} \tag{1}$$

where P_n^{tx} is the transmit power consumption on the nth channel, and P_{cir} is the circuit power.

C. TPR Model

In the CIoT system, in order to enhance the transmission efficiency, the adaptive modulation [31] method is adopted in our system. In this paper, we choose the 2^{η}-quadrature amplitude modulation (QAM). Each device in the system aims to achieve the maximum transmission rate by accessing the high-quality channel and selecting the high modulation levels.

The estimated bit error rate (BER) can be obtained based on the given received SNR and the modulation levels, which is expressed as follows [32]: $\mathrm{BER}_n = 1 - (1 - 2(1 - \eta^{1/2})Q(\sqrt{3\mathrm{SNR}_n/(\eta - 1)}))$ is the Q-function that means the tail distribution function of the standard normal distribution. Given the BER value on the nth channel, the packet error rate can be obtained by $P_n^{\mathrm{per}} = 1 - (1 - \mathrm{BER}_n)^{L_n^{\mathrm{packet}}}$, where L_n^{packet} is the packet size in bits per packet being transmitted successfully on the nth channel.

According to the packet error rate, the successful TPR of the kth packet on the nth channel can be expressed as

$$\begin{aligned} \mathrm{TPR}_{k,n} &= 1 - P_{k,n}^{\mathrm{per}} = (1 - \mathrm{BER}_{k,n})^{L_n^{\mathrm{packet}}} \\ &= (1 - 2(1 - \eta^{1/2})Q(\sqrt{3\mathrm{SNR}/(\eta - 1)}))^{L_n^{\mathrm{packet}}}. \end{aligned} \tag{2}$$

D. Transmit Delay Model

Simply, the transmission delay consists of two kinds of delay [34]: retransmission and handoff [33]. Assuming that one packet needs to be transmitted N_{tx} times in the MAC layer, the packet retransmission delay is computed as $\tau_{\mathrm{tx}}(N_{\mathrm{tx}}) = (\tau_{\mathrm{mac}} + \tau_{\mathrm{data}})(N_{\mathrm{tx}} + 1)$ [35], where τ_{mac} is the processing time of the handshake components in the MAC, and τ_{data} denotes the time needed to transmit the data packet [35]. Then, the average delay for the retransmission of one packet on the nth channel is expressed as

$$\begin{aligned} T_{\mathrm{retrans}} &= \sum_{i=1}^{N_{\mathrm{tx}}} (P_n^{\mathrm{per}})^{i-1}(1 - P_n^{\mathrm{per}})\tau_{\mathrm{tx}}(i-1) \\ &= \sum_{i=1}^{N_{\mathrm{tx}}} (P_n^{\mathrm{per}})^{i-1}(1 - P_n^{\mathrm{per}})(\tau_{\mathrm{mac}} + \tau_{\mathrm{data}})(i-1). \end{aligned} \tag{3}$$

From (3), we can find that the packet retransmission delay is determined by the packet error rate and the number of the retransmission times. Let $N_{\mathrm{tx}}^{\max}$ denote the maximum retransmission times of one packet; then, we can compute the maximum retransmission delay $T_{\mathrm{rtr}}^{\max}$ based on (3).

The delay of the handoff process can be described by the 802.11 Standard [32], [35], and the maximum handoff time of one packet is approximately $T_{\rm hof}^{\max} = 6T_{\rm rtr}^{\max} + \tau_{\rm proc}$, where $\tau_{\rm proc}$ is the scheduling processing time between the ground access point and the station adapter before transmitting the new packets. According to the analysis [35], the average handoff delay of one packet on the nth channel is given by

$$\begin{aligned} T_{\rm hof} = (P_n^{\rm per})^{2N_{\rm tx}}[(P_n^{\rm per})^{4N_{\rm tx}} \times T_{\rm hof}^{\max} + \{1-(P_n^{\rm per})^{4N_{\rm tx}}\} \\ \times (T_{\rm hof}^{\max} + T_{\rm wt})] + \{1-(P_n^{\rm per})^{2N_{\rm tx}}\}[(P_n^{\rm per})^{4N_{\rm tx}} \\ \times (T_{\rm hof}^{\max} + T_{\rm pd}) + \{1-(P_n^{\rm per})^{4N_{\rm tx}}\} \\ \times (T_{\rm hof}^{\max} + T_{\rm pd} + T_{\rm wt})] \end{aligned} \tag{4}$$

where $T_{\rm pd}$ is the processing time of the ground access point sending probe packets before the authentication phase completion, and $T_{\rm wt}$ denotes the processing time of the handover procedure if the packet is lost [35].

According to the above analysis, the average transmission delay of one packet is computed as

$$T_{\rm delay} = T_{\rm rtr} + T_{\rm hof}. \tag{5}$$

E. Throughput Model

One goal of the CIoT system is to maximize the system transmission throughput. We denote the coding rate as ζ, and all packets have the same coding rate. Then, when the kth packet transmitted using the 2^{η_k}-QAM level, the throughput (in bits) per symbol is and the sum transmission throughput (in bits) of the kth packet is $D \times \zeta \times \eta_k$, where D is the number of symbols per packet and we assume all packet have the same number of symbols. Thus, when the CIoT system successfully transmits K packets, we can calculate the total throughput as

$$B = \sum_{k=1}^{K} D \times \zeta \times \eta_k. \tag{6}$$

In the CIoT system, the adaptive modulation scheme can dramatically choose high modulation levels to send more bits per symbol and improve the throughput. However, the high modulation levels need a high received SNR value, and the received signal may fail to be demodulated correctly if the current SNR value cannot satisfy the minimum SNR requirement of the demodulation.

III. Problem Formulation

The transmission scheduling problem can be modeled by using RL with the goal of maximizing the reward of the CIoT system, which can be analyzed by the following subsections.

RL enables to help the CIoT system to exploit the optimal policy to maximize the reward. Generally speaking, in RL, the optimal policy searching process can be modeled as a Markov decision process (MDP), which can be defined as a tuple (S, A, P, r, γ), where S means the state space, A is the action space set, P denotes the transition probability: $P(s_{t+1}|s_t, a_t)$ when the agent takes the from the current state to a new state $a_t \in A$, r is an immediate reward function, and $\gamma \in [0, 1)$ is a discount factor.

1) Agent: It is each active IoT device in the CIoT system.

2) System state: For each device (agent), the system state is defined as $s = \{\chi_{\rm cs}, \chi_{\rm dp}, \chi_{\rm cq}, \chi_{\rm tl}\}$, where $\chi_{\rm cs}$ indicates the channel status (idle or busy), $\chi_{\rm dp}$ shows the channel access priority level, $\chi_{\rm cq}$ denotes the channel quality (SNR), and $\chi_{\rm tl}$ is the traffic load of the selected channel.

3) Action space: We denote $a = \{\beta_{\rm po}, \beta_{\rm sm}, \beta_{\rm am}\} \in A$ as the three types of actions of each agent on the state s after making the decision in terms of the power consumption control ($\beta_{\rm po}$: active or sleep), the spectrum management ($\beta_{\rm sm}$: access or wait or handoff), and the transmission modulation selection ($\beta_{\rm am}$: adaptive modulation order choice).

4) Policy: The policy of each agent is a function that can be deterministic or stochastic [34]. It determines what action to take with a given current state. Let $\pi(a, s)$ denote a current policy: $\pi(a, s) : S \to A$, which is a mapping from the state space to the action space A.

5) Reward: In the CIoT system, the objective of RL is to search the optimal policy to smartly make a transmission scheduling decision with the integration of the environment through optimizing the accumulated reward. However, the formulation of the reward function directly determines the optimal policy that the IoT device makes.

Motivated by the mean opinion score metric [35], we propose a new reward function for the transmission scheduling in the CIoT system, which is expressed as

$$r = \frac{c_1 + c_2 {\rm TPR}_{\rm avera} + c_3 B_{\rm norm}}{1 + c_4 P_{\rm power}^{\rm norm} + c_5 T_{\rm delay}^{\rm norm}} \tag{7}$$

where the variables ${\rm TPR}_{\rm avera}$, $B_{\rm norm}$, $P_{\rm power}^{\rm norm}$, and $P_{\rm power}^{\rm norm}$ are the average TPR, the normalized system throughput, the normalized power consumption level, and the normalized packet transmission delay, respectively; all these four variables are normalized from 0 to 1, so that each feature can be fairly considered into the reward function. Here

$$\begin{aligned} {\rm TPR}_{\rm avera} &= \sum_{k=1}^{K}\sum_{n=1}^{N} \rho_{k,n} {\rm TPR}_{k,n}/(KN) \\ B_{\rm norm} &= B/B_{\rm ideal} \\ P_{\rm power}^{\rm norm} &= \sum_{k=1}^{K}\sum_{n=1}^{N} \rho_{k,n} \frac{P_{k,n}}{P_{\max}}/(KN) \\ T_{\rm delay}^{\rm norm} &= T_{\rm delay}/T_{\max} \end{aligned}$$

where the variable $\rho_{k,n} \in \{0, 1\}$ determines whether the kth packet is transmitted on the channel n. If it takes the value 1, the kth packet is transmitted on channel n. Otherwise, it takes the value 0. $B_{\rm ideal}$ denotes the ideal throughput that the CIoT system achieves. $T_{\max}$ is the maximum transmission delay threshold. $P_{\max}$ is the maximum power consumption threshold. In (7), we can set all the coefficients $c_i, i \in \{1, 2, 3, 4, 5\}$, to be 1 by using the linear regression process [35], [36].

6) Value function: In MDP or RL, each agent has the ability to evaluate and improve a policy based on a value function, where

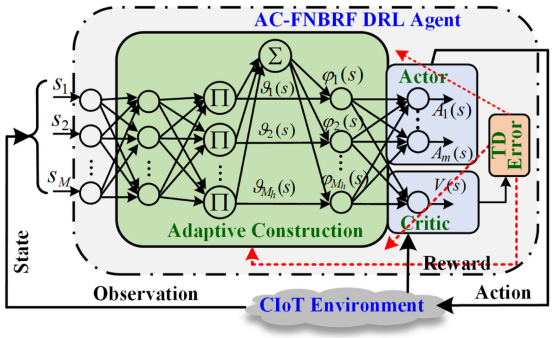

Fig. 2. AC learning framework based on the FNRBF.

the value function can be defined as the expected cumulative discounted reward received over the entire process following the policy. Let $Q^{\pi}(s,a)$ indicate the value function, which is also a cumulative discounted reward at the state with the action under a given policy, and it can be written as

$$Q^{\pi}(s,a) = E\left\{\sum_{t=1}^{\infty}\gamma^t r_t(s_t,a_t)|s_0=s,a_0=a,\pi\right\}. \quad (8)$$

The objective of the transmission scheduling problem is to find a policy π to maximize the sum discounted reward, which can be calculated by using the Bellman optimality equation [14] as

$$J(\pi) = \int_S \gamma^t P(s_t|s_0,\pi)\int_A \pi_\theta(a_t|s_t)Q^{\pi_\theta}(s,a)dads \quad (9)$$

where π_θ is a stochastic policy indicating the state over the current action, with θ being a parameter, and $\pi_\theta(a_t|s_t)$ is the probability density of the action a_t at the state s_t.

From (9), we can see that the optimal policy is achieved iteratively by using the iteration scheme [34] if the system reward $r(s,a)$ and the transition probability $P(s_t|s_0,\pi)$ are known. However, in practical CIoT systems, $P(s_t|s_0,\pi)$ is unknown or partially known; hence, the best policy and optimal reward value $Q^{\pi^*}(s)$ should be learned online by adopting the RL technique.

IV. AC-DRL-Based Intelligent Transmission Scheduling

The optimization problem formulated in Section III can be addressed by using the gradient method, Q-learning, and DRL. However, Q-learning has the slow convergence rate in complex CIoT systems, and it cannot efficiently deal with the continuous state and action space; thus, the learning process will fail miserably. The policy gradient method may converge to a local optimal policy, even if its convergence rate is good. In order to overcome the above-mentioned shortcomings, an AC-FNRBF [37], [38] is proposed to solve the intelligent transmission scheduling problem in CIoT systems; the framework of the proposed AC-FNRBF algorithm can be seen in Fig. 2. In AC, there are two parts, namely, actor and critic, where the actor is used to determine the final action by searching the policy and the critic evaluates the feedback actions by a value function. The AC-FNRBF algorithm is suitable for learning stochastic policies in CIoT systems under high dimensional system state, where the channel condition and the power consumption levels in our formulated optimization problem have continuous spaces.

As shown in Fig. 2, the FNRBF is adopted to achieve the critic value function and the actor policy function without using the Q-function approximator and the prior information. In this structure, the actor aims to provide the random policy that maps the state to the action, and the critic builds the state–action evaluation function based on the current strategy from the actor. The implementation of the actor and the critic through the FNRBF can not only reduce the forward calculation time, but also adaptively respond to the dynamic changes in the unknown dynamic mobile networks. The details of the learning process can be seen in the following subsections.

A. AC Learning Framework Based on the FNBRF

The input of the actor and the critic is the state space from the environment. The output of the FNRBF is the estimated functions of both the actor and the critic. Generally, there are four layers in the AC learning framework based on the FNRBF.

1) First Layer: The system state $s_t = [s_{1,t},\ldots,s_{M,t}]^T \in R^M$ of the CIoT system at the time step t is input into this layer, and each neuron in this layer represents an input state variable $s_{m,t}$. Then, the system state space is directly passed into the input vector of the next layer.

2) Second Layer: It is the hidden layer (rule layer). Each node of the layer represents the front part of a fuzzy rule, and the hidden layer nodes have the following Gaussian function:

$$\vartheta_{ji}(s_t) = \exp\left(-\frac{(s_j - u_{ji}(s_t))^2}{2\sigma_{ji}^2}\right), i = 1,2,\ldots,M_h \quad (10)$$

where $u_{ji}(s_t)$ and σ_j are the mean value (the center) of the Gaussian function and the scalar quantity (the width) of the Gaussian function in the jth hidden layer node.

When the system state is s_t, the fitness of the fuzzy rule of the jth hidden layer node is the product of the Gaussian function, which is given by

$$\Phi_i(s_t) = \prod_{j=1}^{M}\vartheta_{ji}(s_{j,t}) = \exp\left(-\sum_{j=1}^{M}\frac{(s_{j,t}-u_{ji}(s_t))^2}{2\sigma_{ji}^2}\right). \quad (11)$$

3) Third Layer: It is the normalized fitness layer. The function of this layer is to uniformly measure the fitness of each rule and normalize the fitness of all rules. The ith node's corresponding normalized fitness function is

$$\varphi_i(s_t) = \frac{\Phi_i(s_t)}{\sum_{l=1}^{M_h}\Phi_l(s_t)}. \quad (12)$$

4) Last Layer: It is the output layer. The output of the AC-FNRBF is composed of the actor and the critic. The actor network outputs the action function $A_l(s_t)$, and the actor network outputs the value function $V(s_t)$, which can be

expressed as

$$A_l(s_t) = \sum_{i=1}^{M_h} \omega_{ji}\varphi_i(s_t) \tag{13a}$$

$$V(s_t) = \sum_{i=1}^{M_h} \nu_i\varphi_i(s_t) \tag{13b}$$

respectively, where ω_{ji} denotes the weight between the ith node of the hidden layer and the jth output node of the actor network, and ν_i is the weight between the ith node in the hidden layer and the output node of the critic network.

In the actor network, the output action $A_i(s_t)$ cannot be directly used in the CIoT system due the existence of the "exploration-utilization" problem in RL [39]. However, a Gaussian interference can be superimposed on the control actions of the actor network in order to solve the dilemma problem of "exploration-utilization" [39]. We can use the Gaussian interference to achieve the actual action function $\tilde{A}_i(s_t)$ [40].

The temporal difference (TD) error between the estimated value and the real value can be given as

$$\delta_t = r_t + \gamma V_\nu(s_{t+1}) - V_\nu(s_t). \tag{14}$$

The obtained parameters $\tilde{A}_i$, ϖ_i, and ω_{ji} are used to update the center u_i and the width σ_i of the ith hidden layer node in the FNBRF as

$$u_i(t+1) = u_i(t) + \alpha_u\delta_t\left[\frac{\varphi_i(1-\varphi_i)\omega_{ji}(s_t - u_i(t))}{\sigma_i^2}\right] \tag{15a}$$

$$\sigma_i(t+1) = \sigma_i(t) + \alpha_\sigma\delta_t\left[\frac{\varphi_i(1-\varpi_i)\varphi_{ji}(s_t - u_i(t))}{\sigma_i^2}\right] \tag{15b}$$

where α_u and α_σ are the learning rates for the center and the width.

In the critic part, the current action and policy may affect both the current reward and the future rewards in the following time steps; the eligibility trace mechanism corresponding to connect the parameter weight v is used for the updated TD error. The eligibility trace mechanism can improve the learning process and propagate the TD error step forward step by step [16]. Let z_t denote the eligibility trace at the time step t; then, the updated equations for z_t and v_t are written as

$$z_t = \sum_{j}^{t-1}\left((\gamma\lambda)^{t-j}\nabla_{v_i}V_j(s_t)\right) \tag{16a}$$

$$\nu_i(t+1) = \nu_i(t) + \alpha_c\delta_t z_t \tag{16b}$$

where α_c is the learning rate for the weights of the critic network, and $\lambda \in [0, 1]$ is the eligibility trace decay parameter.

In the actor part, at the end of one time stage t, the policy is improved at the actor by using the TD error, and it can be updated as

$$A_l(s_{t+1}) = A_l(s_t) + \alpha_a\delta_t \tag{17}$$

where α_a is a positive parameter of the actor.

In AC, the value function can be achieved in the critic part, which is used to evaluate the quality of the selected action, and the action value function can be obtained in the actor part, which is finally adopted to determine the selection of the action.

B. Structure Update of the AC-FNRBF

In the AC-FNRBF algorithm, the number of hidden nodes is neither given nor fixed, and the nodes can be added or reduced adaptively based on the dynamic changes of the CIoT system. However, there is a key challenge of how to judge whether the number of hidden layer nodes of the FNRBF network needs to be increased or reduced. The increase or decrease in the number of the hidden layer nodes in the FNRBF network depends on the resource allocating network [39], which can adaptively adjust the structure of the hidden layer. Hence, in this subsection, we presented a criterion to adaptively adjust the structure of the AC-FNRBF network in the dynamic CIoT system.

1) Neuron Generation: If the system pattern is not currently well represented by the FNRBF network, then a new unit should be added to reconstruct the pattern. The neural network increases the number of hidden layer nodes if one of the following three conditions occurs.

1) Calculate the minimum distance $d_{\min}$ between the input system state s_t and the center $u_{ji}(s_t)$ of the existing AC-FNRBF network with $d_{\min} = \min||s_t - u_{ji}(s_t)||$. If it is bigger than the effective radius of the boundary ε_d : $d_{\min} > \varepsilon_d$, a new hidden layer node should be added.
2) If the corresponding TD error value δ_t is bigger than the given maximum TD error threshold ε_δ: $\delta_t > \varepsilon_\delta$, a new hidden layer node should be added.
3) If the best fitness of the fuzzy rule is smaller than the given maximum fitness threshold ε_Φ: $\max\Phi_i(s_t) < \varepsilon_\Phi$, a new hidden layer node should be added. The reason is that the current hidden nodes in the rule layer of the FNRBF network fail to effectively cover the input state space.

If one of the above three cases occurs, we can add a new hidden node with

$$\mu_{M_{h+1}} = s_t \tag{18a}$$

$$\sigma_{M_{h+1}} = \kappa d_{\min} \tag{18b}$$

where κ denotes an overlap factor that controls the overlap of responses of the radial basis function units [40].

2) Neuron Combination: In order to simplify the network structure and decrease the computing complexity in the AC-FNRBF network, we can combine the hidden layer nodes if they have the similar functions, that is, if the center value and the width of the two nodes (the node i and the node m) are approximately equal, e.g.,

$$\Delta u_{im} = \sum_{j=1}^{M_h}\left(u_{ji}(s_t) - u_{jm}(s_t)\right)^2 \le \varepsilon_u \tag{19a}$$

$$\Delta\sigma_{im} = \sum_{j=1}^{M_h}\left(\sigma_{ji}(s_t) - \sigma_{jm}(s_t)\right)^2 \le \varepsilon_\sigma \tag{19b}$$

where ε_u and ε_σ are the thresholds. In this case, we consider that the function of these two nodes is similar to each other, and

we can combine these two nodes into a node. The weights can be updated by $\omega_{ji} = (\omega_{ji} + \omega_{jm})/2$ and $\nu_i = (\nu_i + \nu_m)/2$.

3) Neuron Deletion: As the improving learning process, the structure of the AC-FNRBF network will continue to expand. In this case, we can delete some hidden nodes to keep the network structure in the best state. With the increase in the number of the learning steps, the fitness foundations of some hidden nodes will decrease to a certain low level, and these hidden nodes have a small contribution to the learning process. Hence, these nodes can be deleted from the network if its fitness satisfies the following:

$$\frac{\varphi_i(s_t)}{\sum_{l=1}^{M_h} \varphi_l(s_t)} < \varepsilon_\varphi \tag{20}$$

where ε_φ is the minimum fitness threshold. According to the above analysis, the number of the hidden nodes is adaptively changed based on the dynamic system features. Hence, the adaptive AC-FNRBF can efficiently make intelligent decision in the CIoT system.

C. AC-FNBRF for Transmission Scheduling

The AC-FNBRF-algorithm-based intelligent transmission scheduling for CIoT systems is shown in Algorithm 1. The spectrum management, adaptive modulation, and power control can be intelligently updated by using the AC-FNBRF algorithm in the dynamic CIoT system. In CIoT systems, each IoT device can be regarded as an agent, and everything outside the IoT devices constitutes the environment. Each IoT device (agent) observes the system state by interacting with the environment; then, it selects its corresponding action according to the learned policy through the learning framework. In detail, in CIoT systems, at each step, each IoT device (agent) observes the system state s (channel status (idle or busy), channel access priority level, channel quality (SNR), and traffic load of the selected channel); then, it chooses an action a (power consumption level, spectrum management, and transmission modulation selection) based on its policy strategy $\pi_\theta(s, a)$. Then, the CIoT system provides a new system state and the reward r in (7) to agents. Afterward, the proposed AC-FNRBF algorithm is used to approximate both the action function of the actor $A_l(s_t)$ and the value function $V(s_t)$ of the critic simultaneously. Then, the optimal actions for the intelligent transmission scheduling can be searched for CIoT systems in the learning framework.

Algorithm 1: AC-FNBRF for CIoT Systems.

Input: Set the learning rate factor α_a and α_c , discount parameter γ, decay factor λ, all thresholds (e.g., $\varepsilon_d, \varepsilon_\delta$).

1: **Initialize:** Initial state s_0, the initial AC-FNBRF parameters (e.g., $\omega_{ji}(0)$, $\nu_i(0)$), eligibility trace z_0.
2: **For** each time step t=0, 1, 2,... **do**
3: Observe the system state s_t;
4: Obtain the feedback reward r_t;
Search the optimal policy by the AC-FNBRF network
5: Compute the action function: $A_l(s_t) = \sum_{i=1}^{M_h} \omega_{ji}\varphi_i(s_t) + n_i(0, \sigma_V(s_t))$;
6: Compute the value function: $V(s_t) = \sum_{i=1}^{M_h} \nu_i\varphi_i(s_t)$;
7: Compute the TD error: $\delta_t = r_t + \gamma V_\nu(s_{t+1}) - V_\nu(s_t)$;
8: Update the center and the width:

$$\begin{matrix} u_i(t+1) = u_i(t) + \alpha_u\delta_t\varphi_i(1-\varphi_i)\omega_{ji}(s_t - u_i(t))/\sigma_i^2; \\ \sigma_i(t+1) = \sigma_i(t) + \alpha_\sigma\delta_t\varphi_i(1-\varpi_i)\varphi_{ji}(s_t - u_i(t))/\sigma_i^2 \end{matrix};$$

9: Update the eligibility trace: $z_t = \sum_j^{t-1} ((\gamma\lambda)^{t-j}\nabla_{v_i} V_j(s_t))$;
10: Update the weight: $\nu_i(t+1) = \nu_i(t) + \alpha_c\delta_t z_t$;
Structure Update of AC-FNBRF network
11: **If** $d_{\min} > \varepsilon_d$ or $\delta_t > \varepsilon_\delta$ or $\delta_t > \varepsilon_\delta$
Add a new hidden node M_{h+1} with $\mu_{M_{h+1}} = s_t$, $\sigma_{M_{h+1}} = \kappa d_{\min}$;
12: **Else if** $\Delta u_{im} \le \varepsilon_u$ and $\Delta\sigma_{im} \le \varepsilon_\sigma$.
Combine these two hidden nodes into one node with $\omega_{ji} = (\omega_{ji} + \omega_{jm})/2$, $\nu_i = (\nu_i + \nu_m)/2$;
13: **Else if** $\varphi_i(s_t)/\sum_{l=1}^{M_h} \varphi_l(s_t) < \varepsilon_\varphi$
Delete this hidden node;
14: **End if**
15: **End for**

V. Simulation Results and Analysis

In this section, we evaluate the performance of our proposed AC-FNBRF algorithm (called AC-FNBRF) in the CIoT system and compare it with the following algorithms: 1) classical AC RL algorithm based on the policy gradient theorem [13] (denoted as AC-PG; it is also analyzed in Section III-C); 2) deep Q-learning RL algorithm [21] (denoted as DQL); and 3) greedy policy RL algorithm, where the agent exploits the learning strategies with a certain probability (denoted as RL-greedy), explained in [28].

We consider that the wireless IoT devices are randomly distributed in a circular cell area with a radius of 250 m, and each IoT device adaptively chooses the following modulation levels: BPSK, 4-QAM, 8QAM, and 16QAM. The number of available channels is 8, and each channel has eight channel states [16] with the frequency-selective fading model. The coding rate $\zeta = 0.8$. We set all the coefficients to be 1. The maximum transmission delay threshold $T_{\max} = 100$ ms, and the maximum power consumption threshold $P_{\max} = 50$ mW. The circuit power $P_{\text{cir}} = 5$ mW. Each packet size is 1500 bits, and the buffer size is ten packets. The number of time slots is 2000, with the duration of each time slot being $\Delta T = 10$ ms. The background noise power is -90 dBm. The other main simulation parameters are listed in Table I. The neural network has three hidden layers, and the initial number of nodes (neurons) in the hidden layers in the FNBRF network is [15, 20, 15]. The packet transmission follows a Poisson point process.

Fig. 3 presents the learning process of the four algorithms in terms of the TPR, normalized system throughput (norm. throughput), power consumption level, transmission delay, and the reward value when the normalized packet arrival rate is 0.5. We can see that the proposed AC-FNBRF algorithm achieves the optimal policy with a faster convergence rate than the other

TABLE I
SIMULATION PARAMETERS

Parameters	Values	Parameters	Values
Learning rate of the critic α_c	0.04	TD error threshold ε_δ	0.1
Learning rate of the center α_u	0.025	Learning rate of the width α_σ	0.02
Trace decay factor	0.5	Fitness threshold ε_Φ	0.02
The threshold ε_u	0.02	Reward discount factor γ	0.02
The threshold ε_σ	0.04	The boundary ε_d	0.05

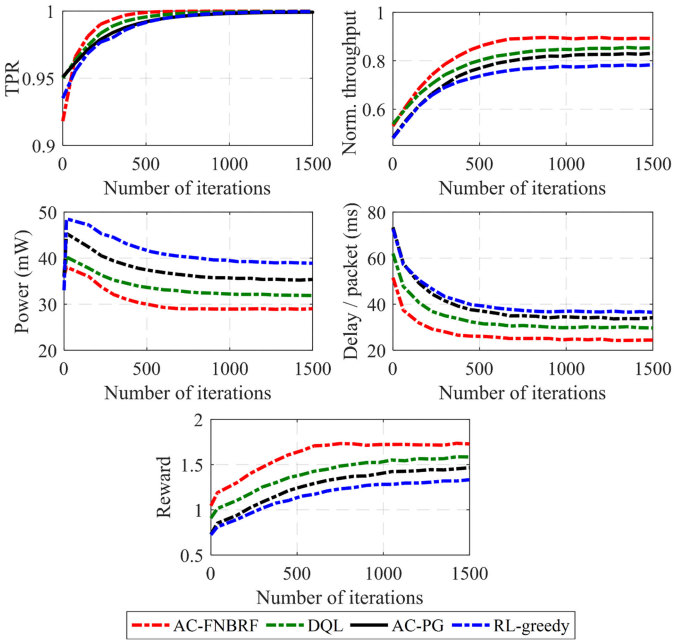

Fig. 3. Learning process of the performance for the four algorithms.

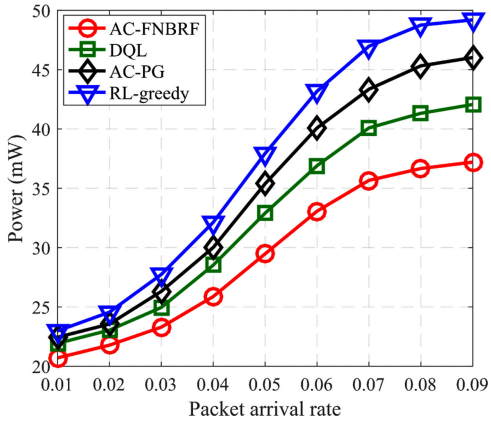

Fig. 4. Average power consumption level versus packet arrival rate.

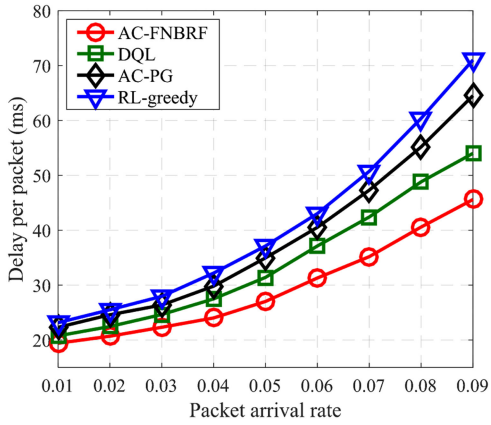

Fig. 5. Average transmission delay per packet versus packet arrival rate.

three algorithms. More specifically, the AC-FNBRF outperforms other algorithms with the higher TPR and system throughput, the lower power consumption and transmission delay, and, thus, the higher reward. Moreover, with the increased number of iteration time steps, the performance gap between the proposed algorithm and other algorithms becomes more obvious, which indicates the efficient leaning process of the AC-FNBRF.

For the DQN algorithm, it needs to search the Q-function approximator, which may fail miserably when the CIoT system is big and complex. And Q-learning cannot efficiently deal with the continuous variables. Thus, its performance is lower than that of our proposed algorithm. In addition, AC-PG achieves the fast convergence speed, but it may converge to the local optimal solution due to the policy gradient method. For the RL-greedy algorithm, its performance is worst among the four algorithms, because it makes the scheduling decision/action only based on the current immediate reward without considering the long-term benefit, but it has a simple control structure. Our proposed algorithm adopts the fuzzy neural network to approximate both the action function and the value function based on the current observed states, and the optimal policy will be learned without using the prior knowledge.

Fig. 4 compares the average power consumption levels of the four algorithms under different normalized packet arrival rates. We can observe that as the packet arrival rate increases, the system needs to consume more power to transmit more packets for the four algorithms. In addition, the increase in the number of retransmission and handoff processes needs extra energy consumption if more transmission packets are needed to be transmitted in the CIoT system. Among the four algorithms, our proposed AC-FNBRF algorithm has the lowest power consumption cost by adaptively adjusting its learning structure based on the dynamic changes in the environment.

The average delay per packet in the CIoT system for the four algorithms is shown in Fig. 5. As the packet arrival rate increases, more transmission packets need to be transmitted. In this case, even if all the RL algorithms attempt to transmit more packets as many as possible through searching the optimal policy, it cannot complete all services. This case may lead to the frequent handover and retransmission. Hence, the transmission delay gradually increases for all algorithms. But the proposed algorithm achieves the best performance; it decreases the transmission delay up to about 10.2, 18.7, and 25 ms, compared

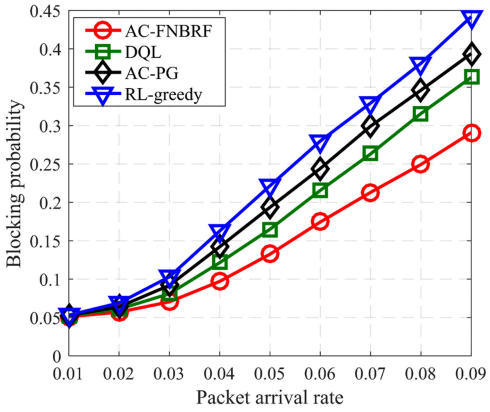

Fig. 6. System blocking probability versus packet arrival rate.

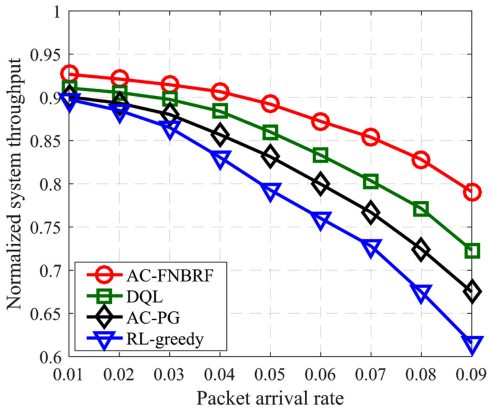

Fig. 7. Normalized system throughput versus packet arrival rate.

with the DQN algorithm, AC-PG algorithm, and RL-greedy algorithm, respectively, when the normalized packet arrival rate is 0.8.

Fig. 6 indicates that the blocking probability of all algorithms improves when the packet arrival rate increases (note that the blocking probability is defined as the ratio of the number of blocked packets and the number of arrived packets [35]). This is because when the system radio resource is fixed under the heavy traffic load, the optimal transmission scheduling decision becomes meaningless if more packets are arriving; thus, the high blocking probability occurs; finally, the growth of the blocking probability is approximately linear as the traffic arrival rate increases. However, the proposed algorithm has better blocking probability performance than other algorithms through efficiently scheduling transmission packets in the CIoT system.

Fig. 7 shows the normalized system throughput of the four algorithms with respect to the packet arrival rate in the CIoT system. We can observe that the normalized throughput decreases when the number of packets increases. The system resource is limited and fixed; when a large number of packets need to be transmitted, it fails to complete all the packet services, which results in bringing down the normalized system throughput. However, the proposed algorithm performs particular suitable and better than other algorithms. The RL-greedy algorithm selfishly

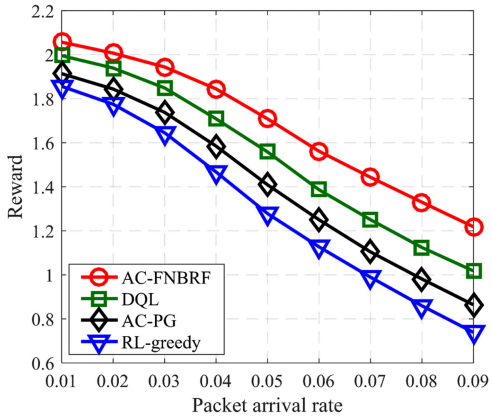

Fig. 8. Reward versus packet arrival rate.

focuses on the current reward without considering the long-term benefit, so it fails to effectively complete the packet transmission, which degrades the system throughput performance.

The reward performance comparison in the CIoT system for the four algorithms is shown in Fig. 8. The reward of all algorithms decreases gradually as the packet arrival rate increases. In the low packet arrival rate regions, the enough radio resource can satisfy the packets transmission, so the reward for all algorithms is high. However, when the packet arrival rate is large, the radio resource may fail to support the packet transmission requirements, leading to the frequent retransmission, handoff, and blocking; all these factors increase the transmission delay and power consumption, as well as decrease the successful TPR and system throughput. Thus, the reward is low in the high packet arrival rate regions. However, our proposed algorithm can still maintain the reward value at a considerable level and achieves better performance than other algorithms, especially in the high packet arrival rate regions.

VI. Conclusion

In this paper, we have proposed an AC-FNRBF algorithm to solve the intelligent transmission scheduling problem in CIoT systems. The proposed algorithm directly approximates both the action function of the actor and the state–action value function of the critic without the system prior knowledge and the approximated Q-function. In the learning framework, we proposed a new reward function that considers the TPR, system throughput, power consumption, and transmission delay. Furthermore, we have analyzed that the proposed AC-FNRBF algorithm has the ability to adjust the learning structure and parameters in dynamic environments. Simulation results verify that the AC-FNRBF algorithm has higher performance than other RL algorithms.

References

[1] J. An, K. Yang, J. Wu, N. Ye, S. Guo, and Z. Liao, "Achieving sustainable ultra-dense heterogeneous networks for 5G," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 84–90, Dec. 2017.

[2] N. C. Luong *et al.*, "Data collection and wireless communication in Internet of Things (IoT) using economic analysis and pricing models: A survey," *IEEE Commun. Surv. Tut.*, vol. 18, no. 4, pp. 2546–2590, Jul. 2016.

[3] Q. Wu *et al.*, "Cognitive Internet of Things: A new paradigm beyond connection," *IEEE Internet Things J.*, vol. 1, no. 2, pp. 129–143, Apr. 2014.
[4] A. Sheth, "Internet of Things to smart IoT through semantic, cognitive, and perceptual computing," *IEEE Intell. Syst.*, vol. 31, no. 2, pp. 108–112, Mar. 2016.
[5] K. Zaheer, M. Othman, M. H. Rehmani, and T. Perumal, "A survey of decision-theoretic models for cognitive Internet of Things (CIoT)," *IEEE Access*, vol. 6, pp. 22489–22512, 2018.
[6] R. Han, Y. Gao, C. Wu, and D. Lu, "An effective multi-objective optimization algorithm for spectrum allocations in the cognitive-radio- based internet of things," *IEEE Access*, vol. 6, pp. 12858–12867, 2018.
[7] H. B. Salameh, S. Almajali, M. Ayyash, and H. Elgala, "Spectrum assignment in cognitive radio networks for Internet-of-Things delay- sensitive applications under jamming attacks," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1904–1913, Jun. 2018.
[8] T. Hassan, S. Aslam, and J. W. Jang, "Fully automated multi-resolution channels and multithreaded spectrum allocation protocol for IoT based sensor nets," *IEEE Access*, vol. 6, pp. 22545–22556, 2018.
[9] N. Abuzainab, W. Saad, C. S. Hong, and H. V. Poor, "Cognitive hierarchy theory for distributed resource allocation in the internet of things," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 7687–7702, Dec. 2017.
[10] A. Aijaz and A. H. Aghvami, "Cognitive machine-to-machine communications for Internet-of-Things: A protocol stack perspective," *IEEE Internet Things J.*, vol. 2, no. 2, pp. 103–112, Apr. 2015.
[11] X. Zhong, R. Lu, L. Li, and S. Zhang, "ETOR: Energy and trust aware opportunistic routing in cognitive radio social internet of things," in *Proc. IEEE Global Commun. Conf.*, Singapore, 2017, pp. 1–6.
[12] L. Zhu, Y. He, F. R. Yu, B. Ning, T. Tang, and N. Zhao, "Communication-based train control system performance optimization using deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10705–10717, Dec. 2017.
[13] K. A. M , 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 1 2018.
[14] Y. Wei, F. R. Yu, M. Song, and Z. Han, "User scheduling and resource allocation in HetNets with hybrid energy supply: An actor-critic reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 680–692, Jan. 2018.
[15] H. Yang, X. Xie, B. Rong, and M. Kadoch, "Intelligent resource management based on efficient transfer actor-critic reinforcement learning for IoV communication networks," *IEEE Trans. Veh. Technol.*, to be published, doi: 10.1109/TVT.2018.2890686.
[16] N. Mastronarde and M. V. D. Schaar, "Fast reinforcement learning for energy-efficient wireless communication," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6262–6266, Dec. 2011.
[17] N. Mastronarde and M. van der Schaar, "Joint physical-layer and system-level power management for delay-sensitive wireless communications," *IEEE Trans. Mobile Comput.*, vol. 12, no. 4, pp. 694–709, Apr. 2013.
[18] P. V. R. Ferreira *et al.*, "Multiobjective reinforcement learning for cognitive satellite communications using deep neural network ensembles," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 5, pp. 1030–1041, May 2018.
[19] X. He, K. Wang, H. Huang, T. Miyazaki, Y. Wang, and S. Guo, "Green resource allocation based on deep reinforcement learning in content-centric IoT," *IEEE Trans. Emerg. Topics Comput.*, to be published, doi: 10.1109/TETC.2018.2805718.
[20] S. Kim, "R-learning-based team game model for Internet of Things quality-of-service control scheme," *Int. J. Distrib. Sens. Netw.*, vol. 13, no. 1, pp. 1–10, 2017.
[21] C. Wang, L. Zhang, Z. Li, and C. Jiang, "SDCoR: Software defined cognitive routing for internet of vehicles," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3513–3520, Oct. 2018.
[22] J. Zhu, Y. Song, D. Jiang, and H. Song, "A new deep-Q-learning-based transmission scheduling mechanism for the cognitive internet of things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2375–2385, Aug. 2018.
[23] Y. Liu, L. Liu, and W. P. Chen, "Intelligent traffic light control using distributed multi-agent Q learning," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst.*, Yokohama, Japan, 2017, pp. 1–8.
[24] Y. Li, K. K. Chai, Y. Chen, and J. Loo, "Smart duty cycle control with reinforcement learning for machine to machine communications," in *Proc. IEEE Int. Conf. Commun. Workshop*, London, U.K., 2015, pp. 1458–1463.
[25] M. Chen, F. Herrera, and K. Hwang, "Cognitive computing: architecture, technologies and intelligent applications," *IEEE Access*, vol. 6, pp. 19774–19783, 2018.
[26] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu, "Context-aware computing, learning, and big data in Internet of Things: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 1–27, Feb. 2018.
[27] H. Li, K. Ota, and M. Dong, "Learning IoT in edge: Deep learning for the Internet of Things with edge computing," *IEEE Netw.*, vol. 32, no. 1, pp. 96–101, Jan. 2018.
[28] B. Xu, C. Yang, and Z. Shi, "Reinforcement learning output feedback NN control using deterministic learning technique," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 635–641, Mar. 2014.
[29] H. R. Berenji, and D. Vengerov, "A convergent actor-critic-based FRL algorithm with application to power management of wireless transmitters," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 4, pp. 478–485, Aug. 2003.
[30] R. Li, Z. Zhao, X. Chen, J. Palicot, and H. Zhang, "TACT: A transfer actor-critic learning framework for energy saving in cellular radio access networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 2000–2011, Apr. 2014.
[31] L. Zhao, H. Wang, and X. Zhong, "Interference graph based channel assignment algorithm for D2D cellular networks," *IEEE Access*, vol. 6, pp. 3270–3279, 2018.
[32] Q. Gao *et al.*, "Robust QoS-aware cross-layer design of adaptive modulation transmission on OFDM systems in high-speed railway," *IEEE Access*, vol. 4, pp. 7289–7300, 2016.
[33] D. Krishnaswamy, "Network-assisted link adaptation with power control and channel reassignment in wireless networks," in *Proc. 3G Wireless Conf.*, 2002, pp. 165–170.
[34] L. Zhu, F. R. Yu, B. Ning, and T. Tang, "Handoff performance improvements in MIMO-enabled communication-based train control systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 582–593, Jun. 2012.
[35] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.
[36] Y. Wu *et al.*, "A learning-based QoE-driven spectrum handoff scheme for multimedia transmissions over cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 11, pp. 2134–2148, Nov. 2014.
[37] G. Bianchi, "IEEE802.11-Saturation throughput analysis," *IEEE Commun. Lett.*, vol. 2, no. 12, pp. 318–320, Dec. 1998.
[38] Q. Zhao, H. Xu, and S. Jagannathan, "Near optimal output feedback control of nonlinear discrete-time systems based on reinforcement neural network learning," *IEEE/CAA J. Autom. Sinica*, vol. 1, no. 4, pp. 372–384, Oct. 2014.
[39] S. Q. Wu and M. Joo Er, "Dynamic fuzzy neural networks—A novel approach to function approximation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 2, pp. 358–364, Apr. 2000.
[40] S. Liu, X. Hu, and W. Wang, "Deep reinforcement learning based dynamic channel allocation algorithm in multi-beam satellite systems," *IEEE Access*, vol. 6, pp. 15733–15742, 2018.

Helin Yang (S'15) is working toward the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore.

His research interests include wireless networks, visible light communication, and resource allocation.

Mr. Yang is a Reviewer for the IEEE COMMUNICATIONS MAGAZINE, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and the IEEE SYSTEMS JOURNAL.

Xianzhong Xie (M'18) received the Ph.D. degree in communication and information systems from Xidian University, Xi'an, China, in 2000.

He is a Professor with the School of Optoelectronic Engineering, and the Director of the Chongqing Key Laboratory of Computer Network and Communication Technology, Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include multiple-input multiple-output precoding, cognitive radio networks, and cooperative communications.