IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON SPECIAL SECTION ON FAIRNESS IN FUTURISTIC WIRELESS NETWORKS ON APPLICATIONS, IMPLEMENTATION, ISSUES, AND OPPORTUNITIES

Received October 24, 2018, accepted November 28, 2018, date of publication December 18, 2018, date of current version January 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886216

Deep Reinforcement Learning Paradigm for Performance Optimization of Channel Observation–Based MAC Protocols in Dense WLANs

RASHID ALI[1], NURULLAH SHAHIN[1], YOUSAF BIN ZIKRIA[1], (Senior Member, IEEE), BYUNG-SEO KIM[2], (Senior Member, IEEE), AND SUNG WON KIM[1]

[1]Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, South Korea
[2]Department of Software and Communications Engineering, Hongik University, Sejong 30016, South Korea

Corresponding author: Sung Won Kim (swon@yu.ac.kr)

This work was supported in part by the Ministry of Science, ICT (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program supervised by the Institute for Information and Communications Technology Promotion (IITP) under Grant IITP-2018-2016-0-00313 and in part by the National Research Foundation of Korea (NRF) funded by the Ministry of Education through the Basic Science Research Program under Grant 2018R1D1A1A09082266.

ABSTRACT The potential applications of deep learning to the media access control (MAC) layer of wireless local area networks (WLANs) have already been progressively acknowledged due to their novel features for future communications. Their new features challenge conventional communications theories with more sophisticated artificial intelligence-based theories. Deep reinforcement learning (DRL) is one DL technique that is motivated by the behaviorist sensibility and control philosophy, where a learner can achieve an objective by interacting with the environment. Next-generation dense WLANs like the IEEE 802.11ax high-efficiency WLAN are expected to confront ultra-dense diverse user environments and radically new applications. To satisfy the diverse requirements of such dense WLANs, it is anticipated that prospective WLANs will freely access the best channel resources with the assistance of self-scrutinized wireless channel condition inference. Channel collision handling is one of the major obstacles for future WLANs due to the increase in density of the users. Therefore, in this paper, we propose DRL as an intelligent paradigm for MAC layer resource allocation in dense WLANs. One of the DRL models, Q-learning (QL), is used to optimize the performance of channel observation-based MAC protocols in dense WLANs. An intelligent QL-based resource allocation (*i*QRA) mechanism is proposed for MAC layer channel access in dense WLANs. The performance of the proposed mechanism is evaluated through extensive simulations. Simulation results indicate that the proposed intelligent paradigm learns diverse WLAN environments and optimizes performance, compared to conventional non-intelligent MAC protocols. The performance of the proposed *i*QRA mechanism is evaluated in diverse WLANs with throughput, channel access delay, and fairness as performance metrics.

INDEX TERMS IEEE 802.11ax, dense WLANs, HEW, reinforcement learning, Q-learning, MAC protocols.

I. INTRODUCTION

Future dense wireless local area networks (WLANs) are attracting significant devotion from researchers and industrial communities. IEEE working groups are expected to launch an amendment to the IEEE 802.11 (WLAN) standard by the end of 2019 [1]. The upcoming amendment, covering the IEEE 802.11ax high-efficiency WLAN (HEW), will deal with ultra-dense and diverse user environments, such as sports stadiums, train stations, and shopping malls. One inspiring service is the promise of astonishingly high throughput to support extensively advanced technologies for 5th generation (5G) communications. HEW is anticipated to infer the various and interesting features of both the learners' environment of a HEW device as well as device behavior in

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

order to spontaneously control the optimal media access control (MAC) layer resource allocation (MAC-RA) [2] system parameters.

In real WLANs, the devices proficiently and dynamically manage WLAN resources, such as the MAC layer carrier sense multiple access with collision avoidance (CSMA/CA) mechanism to improve users' quality of experience (QoE) [3]. Overall device performance depends on exploitation of the instability of network heterogeneity and traffic diversity. Conventionally, the IEEE 802.11 standard uses a binary exponential backoff (BEB) scheme as a CSMA/CA mechanism to avoid collisions [2]. In BEB, a random backoff value is generated from a contention window (CW) to obtain channel access. The CW size is doubled after every collision and reset to its minimum size on successfully transmissions. However, this blindness when increasing and resetting the CW induces performance degradation. For a dense network, resetting the CW to its minimum size may result in more collisions and poor network performance. Likewise, for a small network environment, a blind increase in CW size may cause an unnecessarily long delay while accessing the channel. WLAN resources are fundamentally limited due to shared channel access and wireless infrastructures, whereas WLAN services have become increasingly sophisticated and diverse, each with a wide range of QoE requirements. Thus, for the success of the prospective HEW, it is vital to investigate efficient and robust MAC-RA protocols [2].

Recently, the field of deep learning (DL) has been flourishing in order to enable machine intelligence (MI) capabilities in wireless communications technologies. This newly gained popularity of DL is because of successful applications in different research fields, such as speech recognition, natural language processing, and computer vision. The popular technology titans (Google, Microsoft, Facebook, Amazon, and Nvidia) have already started serious financing of their prevailing computing resources to drive MI research, particularly aiming at DL breakthroughs [5]. DL is now a thriving field in active research topics into relevant applications of wireless communications networks, ranging from learning complex scenarios with unknown channel models to the deployment of cognitive radio networks (CRNs). The use of DL philosophies on the extensive collection of wireless networks has a long history and attained numerous achievements, particularly in the upper communications layers, such as in CRNs and for MAC layer resource management [6]. The WLAN's physical (PHY) layer also poses many challenges for DL [7], such as modulation recognition [8], channel modeling [9], encoding/decoding [10], and channel statistics estimation [11]. It is believed by researchers that WLANs can optimize performance by introducing DL into MAC layer resource allocation. Deep reinforcement learning (DRL) is one DL technique that is motivated by the behaviorist sensibility and control philosophy, where a learner can achieve an objective by interacting with the environment [12]. DRL uses specific learning models, such as the Markov decision process (MDP), the partially observed MDP (POMDP), and Q-learning (QL) [13]. DRL utilizes these techniques in applications like learning an unknown wireless network environment and resource allocation in femto/small cells in heterogeneous networks (HetNets) [13]. Figure 1 depicts RL with its specific learning models and their potential applications in futuristic dense wireless networks.

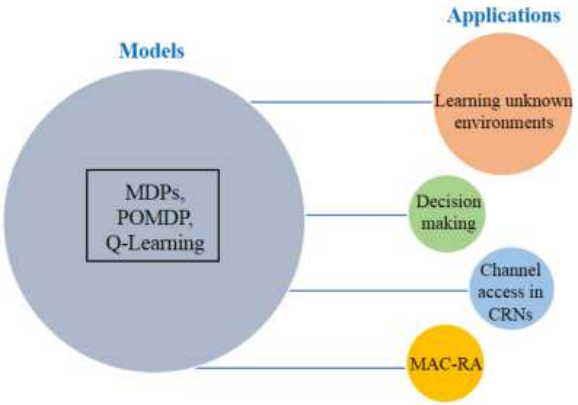

FIGURE 1. Deep reinforcement learning models and their potential applications in dense WLANs.

Motivated by QL, which is one of the prevailing DRL models, we propose an auspicious paradigm for MAC-RA in future dense WLANs. As shown in Figure 2, we envision an intelligent HEW device that accesses channel resources with the assistance of QL techniques, and autonomously observes, learns, and evaluates its actions based on learning in order to achieve optimal performance. QL is inspired by behaviorist psychology, which is used to discover an optimum strategy for taking action for any finite MDP, mainly when the environment is unknown [14]. A significant feature of QL is that it overtly reflects the whole problem of a learner/device interacting with an uncertain environment and being directed to its goal.

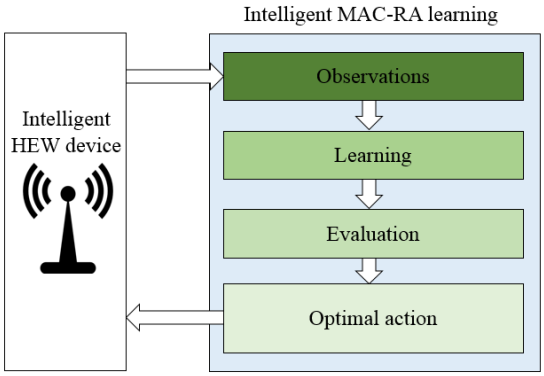

FIGURE 2. Intelligent MAC layer resource allocation (MAC-RA) learning model for an intelligent HEW device.

A goal-directed device can be a tiny piece of a larger behavioral system, such as HEW devices in dense WLAN environments seeking to maximize performance in terms of

throughput and channel access delay. In this paper, we use QL to optimize one of our proposed channel observation–based MAC protocol [15]. In [15], we proposed a channel observation–based scaled backoff (COSB) mechanism to handle the blind increase and reset of CW in BEB. The proposed self-scrutinized COSB adaptively scales up and scales down the backoff CW to enhance the performance of the CSMA/CA in WLANs, specifically in dense environments. Because QL finds solutions through the experience of interacting with an unknown environment, this paper proposes an *intelligent*QL-based resource allocation (*i*QRA) mechanism for performance optimization of COSB.

The rest of this paper is structured as follows. Section II describes deep reinforcement learning in detail along with its challenges and features. This section also highlights the elements, scope, and features of DRL. In Section III, the proposed QL paradigm is defined. This section briefly explains COSB and presents the proposed *i*QRA mechanism in detail. Section IV contains performance evaluation of the proposed mechanism. Finally, comprehensive conclusion and future work are presented in Section V.

II. DEEP REINFORCEMENT LEARNING

In DRL, a device learns the actions to take and maps situations for these actions with the goal of maximizing a numerical reward flag. Usually, a learning device does not know what actions to perform; however, it has to discover which actions produce the best reward by trying them. In many stimulating and inspiring cases, actions might change an instant reward as well as the next reward and, through that, all successive rewards.

DRL is different from traditional supervised DL and unsupervised DL techniques. It is the most recent, focused research in the area of DL. In supervised learning, a learner learns from a given labeled training dataset provided by a knowledgeable external supervisor. This provided dataset describes a situation composed of a description (that is, the label) of the exact action the learner should take in a specific environment. In collaborative problems, it is often impractical to get such datasets of desired behavior that are both correct and representative of all the states in which the learner has to perform actions [14].

DRL is also different from unsupervised learning. Unsupervised learning techniques are about finding structure hidden in collections of unlabeled data. Both supervised and unsupervised learning techniques seem to thoroughly classify DL paradigms. However, in an unfamiliar environment, where one would imagine learning to be most advantageous, a learner must be able to learn from experience.

A. CHALLENGES AND FEATURES OF DRL

The tradeoff between exploration and exploitation is a challenge for DRL that is not in other kinds of learning. To get a considerable reward, a DRL device must learn toward activities attempted in the past and observed to be compelling in creating a reward. In any case, to find such actions, it needs to attempt actions that it has not chosen previously (exploration). The learner needs to exploit what it has effectively experienced, keeping in mind that the target goal is to acquire the maximized reward; however, it also needs to explore in order to make better action selections in the future. The difficulty is that neither exploration nor exploitation can be pursued solely without failing at the task. The learner must attempt an assortment of actions and continuously support those actions that appear to be best. In a stochastic task, each action must be attempted many times to gain a consistent estimate of the expected reward. The exploration–exploitation issue has been intensively examined by mathematicians for a long time, yet remains uncertain [14]. In the HEW system, an intelligent device would exploit to improve its performance, and would explore to know the dynamicity of the WLAN network.

A key component of DRL is that it expressly considers the entire problem of an objective-directed learner interacting with a speculative environment. This is unlike numerous methodologies that consider sub-issues without attending to how they may fit into a bigger picture. DRL takes the opposite strategy, which is beginning with a complete, interactive, objective, seeking learner. All DRL learners have obvious objectives, can detect features of their environments, and can select actions to impact the environments. Besides, it is generally expected from the beginning that the learner still needs to operate, regardless of any huge vulnerability in the environment it faces.

B. ELEMENTS OF DRL

Beyond the agent and the environment, a DRL framework has four primary sub-components: policy (strategy), reward flag, a value function, and, sometimes (optionally), environment model.

1) POLICY

A strategy or policy characterizes the learner's way of acting at a given time. Generally, a policy is a mapping from apparent states of the environment to actions to be taken in those states. It compares to what in psychology would be called a set of action–response relationships. In some cases, the policy might be a straightforward function or lookup table, while in others it might include broad computation (for example, a pursuit procedure). The policy is the essence of a DRL learner in the sense that it alone is adequate to decide its behavior.

2) REWARD FLAG

A reward flag characterizes the objective of a learning problem. At each time step, the environment determines a solitary number called the reward. The learner's main objective is to maximize the total reward it collects in the long run. The reward flag, therefore, expresses the good and bad events for the learner. The reward flag is the essential reason for altering the policy at any state; if an action selected by the policy

brings a low reward, at that point, the policy might be changed to choose some other action for that state in the future.

3) VALUE FUNCTION

While the reward flag shows what is better in an immediate sense, a value function indicates what is best in the end. Thus, the value function of a specific state is the aggregate sum of rewards a learner can collect in the long run, starting from the initial state. For instance, a state may dependably yield a low quick reward, yet at the same time, have a high value function because it is frequently followed by other states that yield high rewards. In any case, it is the value function with which we are most concerned when making and evaluating decisions. Action selections are made based on value verdicts. We pursue actions that bring states of highest values, not the highest rewards, because these actions find the highest extent of rewards for the learner over the long term. In fact, the most significant element of almost all DRL algorithms is a technique for proficiently estimating values.

4) ENVIRONMENT MODEL

An optional component of DRL frameworks is a model of the environment. This is something that mirrors the behavior of the environment, or more generally, that enables suggestions to be made about how the environment will behave. For instance, given a state and an action, the model may anticipate the resultant next state and the next reward. Models are utilized for planning, by which we mean any method for settling on a sequence of actions by considering conceivable future circumstances before they are actually experienced.

C. SCOPE AND LIMITATIONS OF DRL

As discussed above, DRL depends strongly on the notion of the state as input to the policy and the value function. Informally, we can think of the state as a flag passing to the learner with some sense of how the environment is at a specific time. A large portion of DRL techniques are organized around evaluating value functions; however, it is not entirely essential to do this to take care of DRL problems. For instance, approaches like genetic algorithms, genetic programming, simulated forging, and other optimization algorithms have been utilized to approach DRL problems while never engaging value functions [15]. These evolutionary approaches assess the lifetime conduct of numerous non-learners, each utilizing an alternate policy for interfacing with the environment and selecting those actions that are able to acquire the most rewards. If the space of policies is adequately small, or can be organized so that the best policies are common or simple to discover, or if a considerable measure of time is available for the search, then evolutionary approaches can be viable. Furthermore, evolutionary approaches have focal points for problems in which the learner cannot detect the entire state of the environment. In contrast to evolutionary approaches, DRL techniques learn while interfering with the environment. Techniques ready to exploit the details of individual behavioral interactions can be substantially more productive than evolutionary strategies in many types of wireless network.

D. Q-LEARNING MODEL

QL might be summoned to trace an optimal action policy for any given (finite) MDP, particularly for an obscure system model, as presented in Figure 3. In such a case, the QL model is likewise comprised of a learner, of a set of states, S, and a set of actions, A, for every state. By performing an action in a particular state, the learner collects a reward with the objective of maximizing its accumulated reward. Such a reward is represented by a Q-function (also known as a Q-value function). The Q-value is updated in an iterative way after the learner performs an action and observes the resultant reward, as well as the related prospective states, at each time instant [16]. QL has recently been applied in heterogeneous wireless networks. A heterogeneous, completely distributed, multi-objective approach based on an DRL model was developed for self-optimization of femtocells in [17]. That proposed paradigm is supposed to solve both the resource allocation and interference coordination issues in the downlink of femtocells.

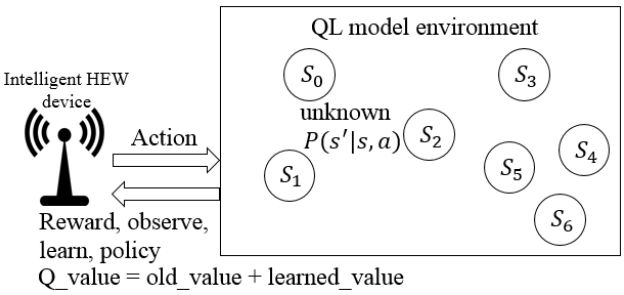

FIGURE 3. Q-learning model environment for an intelligent HEW device.

III. PROPOSED Q-LEARNING PARADIGM FOR DENSE WLANS

In this section, we propose DRL as an auspicious paradigm for channel observation–based MAC protocols in dense HEW networks. This section is further divided into three subsections. The first subsection elaborates one of the channel observation–based MAC protocols, COSB. In the second subsection, we design a QL-based intelligent mechanism (*i*QRA) to optimize the performance of COSB. Third sub-section elaborates the computational complexity of the proposed *i*QRA mechanism.

A. CHANNEL OBSERVATION–BASED MAC PROTOCOL

To unravel the performance deprivation problem in dense WLANs caused by CSMA/CA of conventional MAC layer distributed coordination function (DCF), a more versatile channel observation–based scaled backoff approach is proposed in [3], which primarily relies on the density of the network. The proposed COSB protocol guarantees high throughput and low channel access delay by reducing the

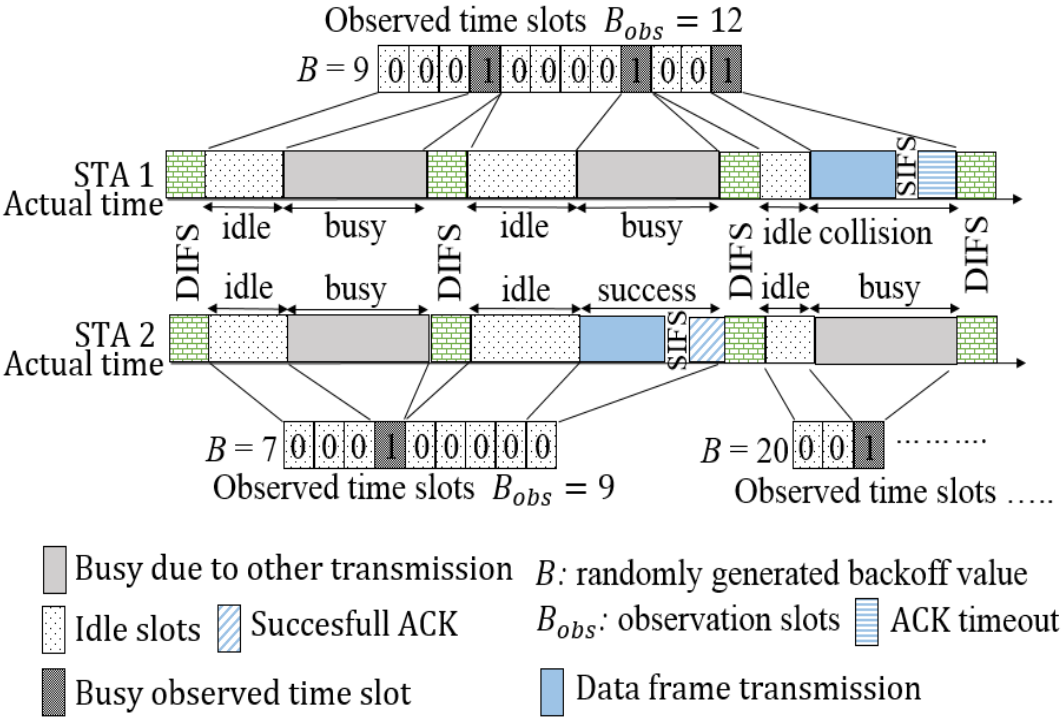

FIGURE 4. Channel observation mechanism of the channel observation–based scaled backoff during the backoff procedure [3].

number of collisions during the channel access procedure. In COSB protocol, the contending stations (STAs) proceed to the backoff procedure by selecting random backoff value B, as shown in Figure 4 ($B = 9$ for STA1, and $B = 7$ for STA2), after the communication medium has been idle for a distributed inter-frame space (DIFS) period. The time immediately following the DIFS is considered as discretized observation time slots (η). The duration of η is either an idle slot time, σ (a constant), or a variable occupied slot time (that is, occupied due to successful transmission or a collision). The value of B decrements by one whenever the medium is detected as idle for σ. A data frame is transmitted after B reaches zero. Furthermore, when the communication channel is detected as occupied, the tagged STA stops decrementing B and continues sensing the channel until it is again sensed as idle for a DIFS period. Every individual contending STA can capably measure the conditional channel collision probability, p_{obs}, which is defined as the probability that a transmission will fail. Subsequently, COSB discretizes the time in B_{obs} observation time slots, where the value of B_{obs} is the total number of η slotted observation slots between two consecutive backoff stages, as presented in Figure 4. A tagged contending STA updates p_{obs} from B_{obs} as follows:

$$p_{obs} = \frac{1}{B_{obs}} \times \sum_{k=0}^{B_{obs}-1} S_k \tag{1}$$

where for observation time slot k, $S_k = 0$ if η is sensed as idle or the tagged STA transmits the data frame successfully, whereas $S_k = 1$ if η is detected as occupied or the tagged STA experiences a collision, as shown in Figuere 4. Instead of resetting the CW after a successful transmission, COSB decrements it exponentially based on the currently measured p_{obs}. Because the current backoff stage represents the number of collisions or successful transmissions of a tagged STA, the increment or decrement of CW is performed as follows:

$$CW_{cur} = \begin{cases} 2 \times CW_{pre} \times \omega^{p_{obs}}, & if\ collision \\ \dfrac{CW_{pre}}{2} \times \omega^{p_{obs}}, & if\ succesful \end{cases} \tag{2}$$

where ω is used as a constant design parameter to control the optimal size of the current CW and is expressed as $\omega = CW_{min}$.

B. INTELIGENT QL–BASED RESOURCE ALLOCATION

The proposed *i*QRA mechanism considers backoff stages as an available set of states, where a learning STA scales up (increments to the next state) and scales down (decrements to the previous state) the size of the CW. An action a, in a particular state s, obtains a reward r, with the aim to exploit its accumulated Q-value function, $Q(s, a)$. This Q-value function is updated in an iterative manner after the STA performs an action and perceives the resulting reward. Figure 5 depicts a model environment of a channel observation–based backoff mechanism (that is, COSB) with its elements for the proposed *i*QRA mechanism. Let $S = \{0, 1, 2, \ldots, m\}$ denote a finite set of m possible states of a HEW environment for the COSB

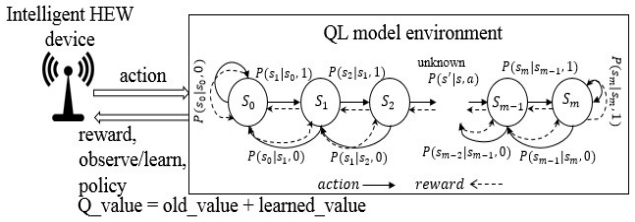

FIGURE 5. ***Intelligent*** **Q-learning–based resource allocation (*i*QRA): system environment and its elements.**

mechanism, and let $A = \{0, 1\}$ represents a fixed set of allowable actions to be taken, where zero indicates a decrement (for successful transmission) and 1 indicates an increment (after a collision). At time slot t, STA observes its current state (s), i.e.$s_t = s \in S$, and takes an action (a), i.e. $a_t = a \in A$. Action a_t changes the state of the environment from s_t to $s_{t+1} = s' \in S$. The main goal of the QL algorithm is to learn an optimal policy that exploits the total anticipated reward, which is given by following Bellman's equation [5],

$$Q^{opt}(s_t, a_t) = \mathbb{E}\{r_t + \beta \times \max_{a'} Q^{opt}(s', q')|s_t = s, a_t =\}. \tag{3}$$

Since the reward may effectively get unbounded, a discounted reward factor, β ($0 < \beta < 1$), is utilized. In the QL algorithm, $Q(s, a)$ estimates the reward as the aggregate reward and is updated as follows:

$$Q(s, a) = (1 - \alpha) \times Q(s, a) + \alpha \times \Delta Q(s, a), \tag{4}$$

where α is the learning rate, defined as $0 < \alpha < 1$. The learning occurs quickly based on improved learning estimate$\Delta Q(s, a)$, and is expressed as

$$\Delta Q(s, a) = \left\{r(s, a) + \beta \times max_a Q\left(s', a'\right)\right\} - Q(s, a). \tag{5}$$

As characterized before, β is the discount rate. Parameter β weighs instant rewards more vigorously than future rewards. The expression $max_a Q(s\prime,a\prime)$ in (3) and (5) defines the best-estimated value for the potential state$s\prime$. In the long run, $Q(s, a)$ converges to the optimal Q-value $Q^{opt}(s, a)$, that is, $\lim_{t\to\infty} Q(s, a) = Q^{opt}(s, a)$. The naivest policy for action selection can be to pick one of the actions with the maximum measured Q-value (exploitation). If there is more than one action with the maximum Q-value, a random choice can be made. This exploitation method is known as a greedy action a^{opt} selection method, and can be written as

$$a^{opt} = argmax_a Q(s, a) \tag{6}$$

where $argmax_a$ represents the exploitation of $Q(s, a)$ with respect to a. The instant reward is maximized by continuous exploitation in a greedy manner. A modest substitute is to exploit more often, but occasionally, the learning STA explores all the allowable actions independent of a^{opt} with probability ε (known as exploration). The greedy and non-greedy selection of actions is known as the ε-greedy method [5]. A feature of the ε-greedy technique is that, as the number of instances increases, every action guarantees the convergence of $Q(s, a)$ to $Q^{opt}(s, a)$. In a HEW environment, a STA would exploit to improve throughput, and would explore to know the dynamicity of the WLAN environment. To balance exploitation and exploration under the proposed *i*QRA mechanism, and ε-greedy method is applied with probability ε for exploration and probability $1-\varepsilon$ for exploitation.

We express the reward in order to minimize channel collision probability p_{obs}. The reward given by action a_t taken at state s_t in slot-time t is expressed as

$$r_t(s_t, a_t) = 1 - p_{obs} \tag{7}$$

The above statement indicates how pleased an STA was with its action in state s_t. Figure 5 depicts the state transition diagram of the *i*QRA mechanism. In the figure, the STA moves from one state to another state with $1-p_{obs}$ as a reward. The STA observes and learns the environment to optimize the backoff process. Algorithm 1 depicts the steps performed by the proposed *i*QRA mechanism to optimize the COSB protocol.

Algorithm 1 COSB Performance Optimization Using *i*QRA

1: **Global initialization:** //The reward and Q-value matrices are initialized globally to keep track of the instant reward and cumulative reward for all possible state transitions (actions) for s states, that is, r (s, a) and Q (s, a).
2: **Function** Select CW using *i*QRA (p_{obs})
Input: channel observation–based collision probability p_{obs}
Output: CW: return optimized contention window
3: **Initialize:** cur_rew = 0, $\Delta Q(s, a) = 0$, $\varepsilon = 0$
4: Calculate reward as cur_rew = 1 -p_{obs}
5: Update reward table for r (s, ra) = cur_rew
6: Calculate improved estimate ΔQ (s,a)according to Equation (5)
7: Update Q-value table for Q (s, a) according to Equation (4)
8: Pick a random value to explore or exploit (ε-greedy method)
9: **if** (exploit)
10: Find a^{opt} according to Equation (6)
11: Scale CW according to the optimal action.
12: **else** (explore)
13: Scale the CW using COSB mechanism
14: **end if**
15: return CW
16: **end Function**

C. COMPUTATIONAL COMPLEXITY

The computational complexity of the proposed *i*QRA mechanism is based on the learning phase of the system. An STA learns the system by exploring different permissible actions in every specific state. However, as soon as the environment

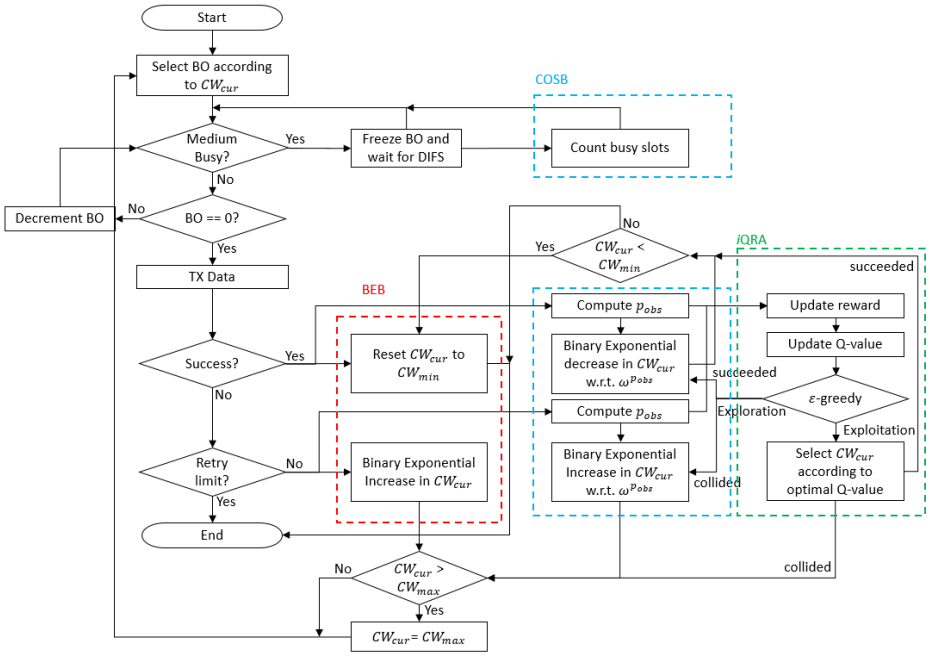

FIGURE 6. CSMA/CA flowchart representing functional comparison of BEB, COSB and *i*QRA mechanisms.

is learned, the best action can be exploited in any given state in an ε-greedy manner, resulting in the optimal solution. Since *i*QRA performs only a constant amount of computation (a fixed number of actions and states), its computational complexity per iteration can be written either as $O(1)$ if explores, or as$O(mln(i))$ for $i \in (1, m)$ of m number of states if exploits. The best case for the computational complexity is when there is only one possible state to move at any state, that is$m = 1$, and the worst case arises with the m number of states. The computational complexity of *i*QRA mechanism is checked for $m = 6$, which is a default value of number of backoff stages in most of the IEEE 802.11 standards. The obtained results remain below $0.000ns+$. Figure 6 shows flowchart of CSMA/CA representing the functional comparison of BEB, COSB and *i*QRA algorithms. The figure helps to understand the addition of functions to the currently implemented CSMA/CA mechanism. An observation-based intelligence is embedded to the CSMA/CA for performance optimization.

IV. PERFORMANCE EVALUATION

We simulated the proposed learning-based *i*QRA mechanism using the ns-3 network simulator, version 3.28 [19], with an IEEE 802.11ax HEW model for dense WLANs. Some important simulation parameters are given in Table 1.

A. QL PARAMETER SELECTION

To evaluate the QL parameters for the proposed *i*QRA, we simulated 25 contending STAs for 100 seconds, varying α and β with small (0.2), medium (0.5) and large (0.8) values. Probability ε was set to 0.5 for balanced exploration and exploitation. Figure 7 shows the convergence of learning estimate ΔQ from Equation (5) with respect to the learning rate (α). The figure depicts how a smaller α makes ΔQ converge faster. The convergence of ΔQ indicates that the STA has learned its environment and can exploit optimal actions in the future. An interesting observation is that ΔQ is not steady in the beginning, which is due to the initial exploration of the environment. Therefore, most of the states do not optimize the value function in the beginning. Later, the STA infers the states that can deliver the most rewards, increasing the cumulative reward. After enough instances (such as 13 instances for $\alpha = 0.2$ in Figure 7), we can see that the learner has found configurations that can lead to optimization of the process. Similarly, we observe in Figure 8 ΔQ converges faster for a

TABLE 1. MAC layer and PHY layer simulation parameters

Parameter Type	Value
Frequency	5 GHz
Channel bandwidth	160/20 MHz
Data rate (MCS11)	1201/54 Mbps
Payload size	1472 bytes
Transmission range	10/25 m
Contention window minimum	32
Contention window maximum	1024
COSB design parameter (ω)	32
Simulation time	100/500 sec
Station position	Fixed/Random
Propagation loss model	LogDistancePropagation
Mobility model	ConstantPositionMobility
Rate adaptation model	ConstantRateWifiManager MinstrelWifiManager
Error rate model	NistErrorRateModel YansErrorRateModel

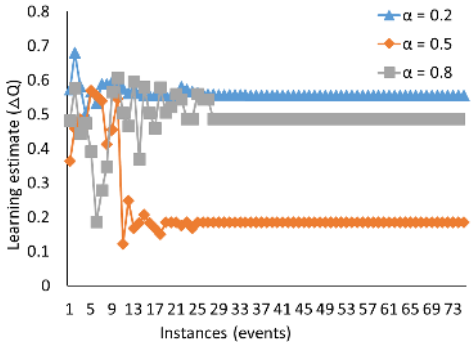

FIGURE 7. Convergence of learning estimate (ΔQ) for varying the learning rate, $\alpha(\beta = 0.8, \varepsilon = 0.5)$.

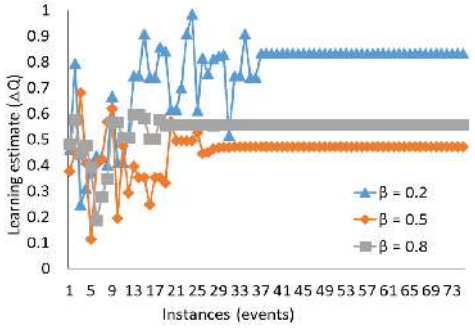

FIGURE 8. Convergence graphs of the learning estimate (ΔQ) from varying the discount factor $\beta(\alpha = 0.2, \varepsilon = 0.5)$.

large value of discount factor β. In both cases (Figure 7 and Figure 8), ε was set to 0.5, indicating equal opportunities for exploration and exploitation. The small value for α and the large value for β (along with equal probability ε) yield the best results for optimization in the system. The convergence of learning estimates shows that an optimal solution for the environment exists.

Figure 9 and Figure 10 portray the effects of the parameters on throughput of the system (Figure 9 for a small network of 15 STAs, and Figure 10 for a dense network of 50 STAs). As shown in Figure 9(a), if ε is set to 0.2 for a small network of 15 STAs, $\alpha = 0.5$ and $\beta = 0.2$ give the best results. However, in this case, decreasing α (that is$\alpha = 0.2$) has little effect on throughput, but increasing it to $\alpha = 0.8$ degrades throughput dramatically. Figure 9(b) shows that if ε and α are set to 0.5, β can be set small, medium, or large. However, for $\varepsilon = 0.8$ and $\alpha = 0.5$, seting β to its medium value ($\beta = 0.5$) enhances throughput, as shown in Figure 9(c). Figures 10(a), 10(b) and 10(c) show that for a dense network system of 50 STAs, a small value for α (that is, $\alpha = 0.2$) and a large value for β (that is, $\beta = 0.8$) are efficient for small and medium values of ε (that is$\varepsilon = 0.2$ and $\varepsilon = 0.5$). With a large value for ε (that is,$\varepsilon = 0.8$), as shown in Fig. 10(c), throughput is improved if the large α and β are used (that is,$\alpha = 0.8$, and $\beta = 0.8$). Thus, from Figure 9 and Figure 10, we show that a combination of smallα, large β, and a medium value for ε (that is, $\alpha = 0.2$, $\beta = 0.8$, and $\varepsilon = 0.5$) is somewhat efficient for both sparse and dense network systems.

B. THROUGHPUT

To evaluate the performance of *i*QRA, we compared simulation results with the traditional binary exponential backoff (BEB) and COSB algorithms. Figure 11 shows how the *i*QRA mechanism optimizes the throughput of COSB, specifically in a dense network of 50 contending STAs. The performance improvement clearly indicates that the QL-based proposed mechanism is effective at learning the wireless network. In a network of five contending STAs, *i*QRA achieves relatively lower system throughput than COSB. The performance of *i*QRA may degrades in small networks due to low and irregular rewards.

C. CHANNEL ACCESS DELAY

The channel access delay for a successfully transmitted data frame is defined as the interval from the time the frame is at the head of the queue (ready for transmission) until successful acknowledgement that the frame was received. If a frame reaches the given retry limit, it is dropped, and its time delay is not included in the calculation of channel access delay. Figure 12 depicts the performance of the proposed *i*QRA mechanism along with the conventional BEB and the original COSB mechanisms in terms of channel access delay (in milliseconds). From the figure, we observe that the proposed *i*QRA mechanism has a higher channel access

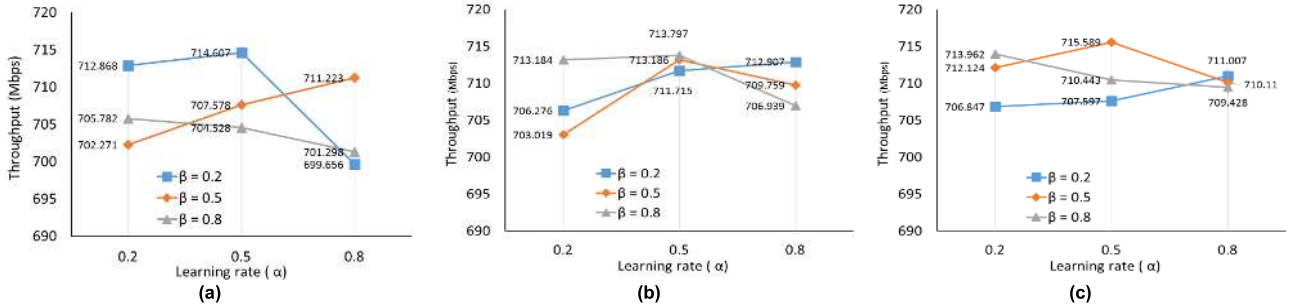

FIGURE 9. Throughput comparison of α and β in a small network of 15 STAs with (a) $\varepsilon = 0.2$, (b) $\varepsilon = 0.5$ and (c) $\varepsilon = 0.8$.

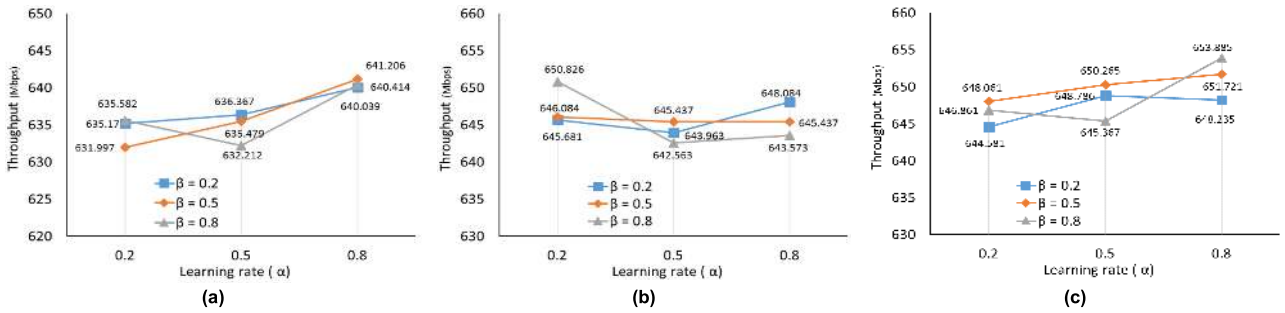

FIGURE 10. Throughput comparison of α and β in a dense network of 50 STAs with (a) $\varepsilon = 0.2$, (b) $\varepsilon = 0.5$ and (c) $\varepsilon = 0.8$.

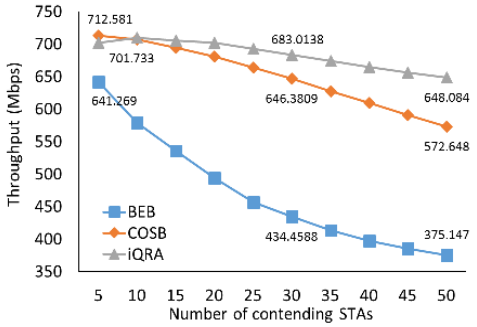

FIGURE 11. Throughput comparison of BEB, COSB, and *i*QRA with $\alpha = 0.2$, $\beta = 0.8$ and $\varepsilon = 0.5$ in a network of five to 50 contending STAs.

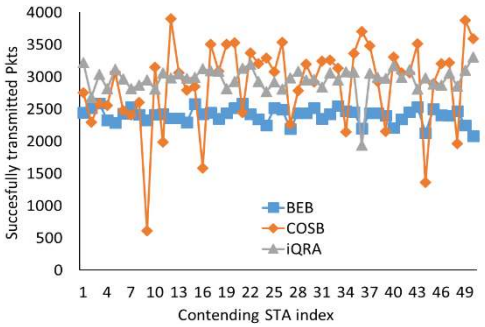

FIGURE 13. The number of successfully transmitted packets by each STA in a dense network of 50 STAs.

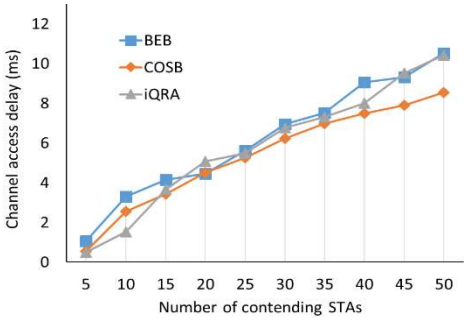

FIGURE 12. Channel access delay comparison of BEB, COSB, and *i*QRA with $\alpha = 0.2$, $\beta = 0.8$, and $\varepsilon = 0.5$ in a network of five to 50 contending STAs.

delay, compared to COSB; however, it does not exceed the conventional BEB mechanism. It is obvious that the *i*QRA mechanism has an increased channel access delay due to its environment inference characteristics.

D. FAIRNESS

The fairness issue can be seen for COSB in Figure 13. In a dense network environment of 50 STAs, COSB suffers from the fairness problem due to some STAs continuously operating at a higher CW size, and a few fortunate STAs can operate at a lower CW size. Under COSB, once the STA reaches a larger CW, it has to transmit successfully many times to return to the smaller CW, which seems difficult in a dense network environment due to the high probability of collision.

The proposed *i*QRA brings fairness to the contending STAs, because every STA autonomously and intelligently exploits its environment. Table 2 shows the values in Jain's fairness index [18] achieved by BEB, COSB, and *i*QRA for a small network of five STAs to a large, dense network of 50 STAs. We observe that the previously proposed COSB mechanism was unfair for small to large network environments, while the *i*QRA mechanism optimizes COSB to perform fairly among the contending STAs, whether it is for a small network or a large network.

TABLE 2. Jain's fairness index comparison.

STAs	BEB	COSB	*i*QRA
5	0.999	0.953	0.999
10	0.999	0.999	0.999
15	0.999	0.999	0.999
20	0.999	0.997	0.999
25	0.999	0.992	0.999
30	0.998	0.993	0.999
35	0.998	0.991	0.999
40	0.997	0.990	0.999
45	0.998	0.948	0.999
50	0.998	0.953	0.998

E. NETWORK DYNAMICITY

Subsequently, QL is essentially intended to make intelligent adjustments according to the dynamics of the environment. A dynamic environment can be the activation of more

contending STAs in the network or the deactivation of previously active STAs. We evaluated the performance of the proposed *i*QRA mechanism by activating five more contending STAs every 50 seconds until the number of STAs reached 50. Figure 14 explains the effects of network dynamics on ΔQ (that is, learning estimate) of a tagged STA. The figure shows 1400 learning instances (events) of a tagged STA during the simulation period (500 sec). Each instance represents the updated value of learning estimate ΔQ whenever a packet transmission is attempted. As shown in the figure, with changes in the number of contending STAs within the network, the tagged STA experiences a fluctuation inΔQ, indicating the change in the environment. Later, this QL-equipped, intelligent tagged STA converges and is capable of optimizing the performance in a dynamic wireless environment. In Figure 15, we see that *i*QRA eventually reaches a steady state in system throughput. On the other hand, BEB and COSB are severely affected by the increase in the number of competing STAs.

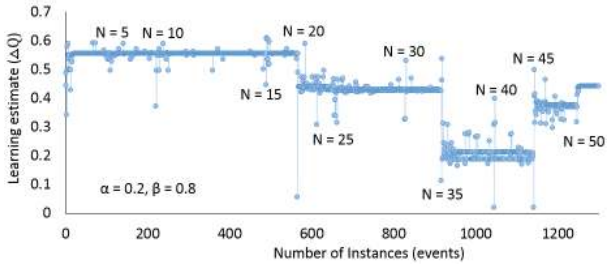

FIGURE 14. Convergence of the learning estimates (δQ) in a dynamic network environment (increasing the number of contenders every 50 seconds).

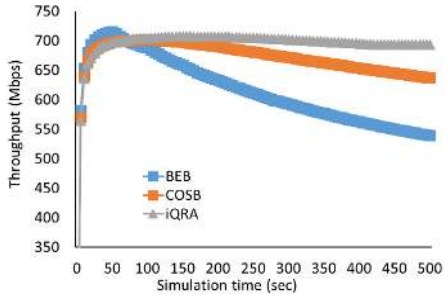

FIGURE 15. System throughput comparison in a dynamic network environment (increasing contenders by five every 50 seconds).

F. DISTANCE-BASED RATE ADAPTATION MODELS

Throughput shown in Figure 11 and Figure 15 are achieved in a network environment using the ConstantRateWifiManager rate-adaptation algorithm [19], in which contending STAs are placed at a fixed distance from the access point (AP). Hence, all the devices are transmitting at a constant data rate. To evaluate the performance of the proposed *i*QRA algorithm, we simulated a more practical and real network environment, such as MinstrelWifiManager [19]. The Minstrel rate adaptation varies the transmission rate of the sender STA to match the WLAN channel conditions (mainly based on the distance from the AP), in order to achieve the best possible performance. The results shown in Figure 16 are achieved in an IEEE 802.11a (11 Mbps) wireless network for $N = 10$. All contending STAs were randomly placed within a distance of 25m from the AP. A tagged STA (initially placed at a 1m distance) moves away from the AP after a step-time of 1sec. Throughput shown in Figure 16 was obtained after each 5m distance from the AP. The performance of a tagged STA for all three of the compared algorithms (BEB, COSB, and *i*QRA) degrades as the distance from the AP increases, as shown in Figure 16. We observe that the throughput of the tagged STA for BEB is close to zero after the STA reaches a distance of 60m, and finally becomes zero when it exceeds the coverage (80m). Under COSB, due to its observation-based nature, a STA achieves higher throughput even after a 60m distance, compared to BEB. However, the proposed *i*QRA maintains performance, even if the distance increases to 80m, due to its intelligence capability.

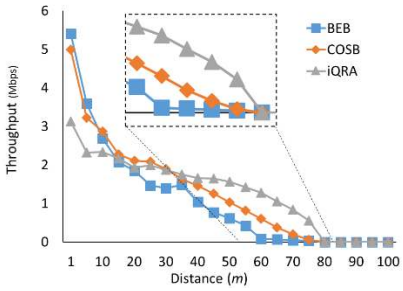

FIGURE 16. Throughput comparison for distance-based rate-adaptation network environments.

G. EFFECTS OF CHANNEL ERROR-RATE MODELS

In order to achieve reliable results to compare with real device performance, it is essential to represent the PHY layer of the WLAN as correctly as possible in simulations. The ns-3 simulator states two error-rate models for calculation of the bit error rate (BER) and corresponding packet error rate (PER): YansErrorRateModel and NistErrorRateModel [19]. Currently, ns-3 recommends using NistErrorRateModel as the default, specifically for ideal channel cases. There is not much difference between these two, except that YansErrorRateModel uses overly optimistic (analytical) results. In Figure 17, we evaluate the effect of the above-stated error-rate models. The figure shows that there is not much

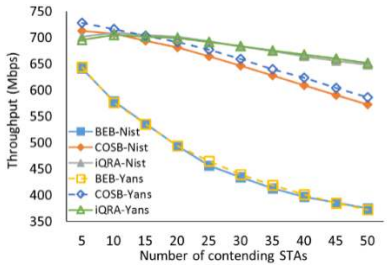

FIGURE 17. Throughput comparison for NistErrorRateModel and YansErrorRateModel network environments.

difference among BEB, COSB, and *i*QRA performance when simulated with the two different error-rate models. The performance of COSB increases a little with YansErrorRateModel. The reason is that, similar to YansErrorRateModel, COSB scales its parameters based on analytical results, that is, channel collision probability. On the other hand, the performance of *i*QRA remains almost the same, because it is the optimized form of COSB.

V. CONCLUSION

The upcoming dense high-efficiency WLAN (that is, IEEE 802.11ax HEW) promises per-device throughput performance that is four times higher. One of the bottlenecks for this performance achievement is tackling the huge challenge of efficient MAC layer resource allocation in WLANs due to their distributed contention-based nature. Currently, a CSMA/CA-based WLAN uses a binary exponential backoff mechanism, which blindly increases and decreases the contention window after collisions and successful transmissions, respectively. To handle the performance degradation challenge caused by the increasing density of WLANs, a self-scrutinized channel observation–based scaled backoff (COSB) mechanism based on a practical channel collision probability was proposed. COSB overcomes the limitation of BEB to achieve high efficiency and robustness in highly dense networks, and enhances the performance of CSMA/CA in dense networks. However, to satisfy the diverse requirements of such dense WLANs, it is anticipated that prospective WLANs will autonomously access the best channel resources with the assistance of sophisticated wireless channel condition inference. Motivated by the potential applications and features of deep reinforcement learning in wireless networks, such as the deployment of cognitive radio, we introduced DRL as a paradigm for MAC layer resource allocation in dense WLANs. In this paper, we propose one of the DRL techniques, Q-learning, as an intelligent paradigm for MAC layer resource allocation in dense WLANs. The proposed DRL paradigm uses intelligent QL-based inference to optimize the performance of COSB, and we call it *intelligent* QL–based resource allocation. Simulation results show that the proposed *i*QRA optimizes the performance of COSB in fixed wireless STA network environments, as well as for randomly placed and distance-based rate adaptation network environments.

Future research considerations include the formulation of a mathematical model for the proposed *i*QRA mechanism. Future work also includes performance evaluations of *i*QRA in more realistic channel-error and signal-to-noise ratio (SINR)–based data rate models.

REFERENCES

[1] IEEE802.org. (2018). *IEEE P802.11—TASK GROUP AX*. Accessed: Feb. 30, 2018. [Online]. Available: http://www.ieee802.org/11/Reports/tgax_update.htm
[2] R. Ali, S. W. Kim, B. Kim, and Y. Park, "Design of MAC layer resource allocation schemes for IEEE 802.11ax: Future directions," *IETE Tech. Rev.*, vol. 35, no. 1, pp. 28–52, 2016, doi: 10.1080/02564602.2016.1242387.
[3] R. Ail, N. Shahin, R. Bajracharya, Y. T. Kim, B.-S. Kim, and S. W. Kim, "A self-scrutinized backoff mechanism for IEEE 802.11ax in 5G unlicensed networks," *Sustainability*, vol. 10, no. 4, p. 1201, 2018, doi: 10.3390/su10041201.
[4] N. Kato *et al.*, "The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 146–153, Jun. 2017, doi: 10.1109/MWC.2016.1600317WC.
[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org
[6] U. Challita, L. Dong, and W. Saad. (2017). "Proactive resource management in LTE-U systems: A deep learning perspective." [Online]. Available: https://arxiv.org/abs/1702.07031.
[7] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017, doi: 10.1109/CC.2017.8233654.
[8] A. Fehske, J. Gaeddert, and J. H. Reed, "A new approach to signal classification using spectral correlation and neural networks," in *Proc. IEEE Int. Symp. New Frontiers Dyn. Spectr. Access Netw. (DYSPAN)*, Nov. 2005, pp. 144–150.
[9] M. Ibukahla, J. Sombria, F. Castanie, and N. J. Bershad, "Neural networks for modeling nonlinear memoryless communication channels," *IEEE Trans. Commun.*, vol. 45, no. 7, pp. 768–771, Jul. 1997.
[10] J. Bruck and M. Blaum, "Neural networks, error-correcting codes, and polynomials over the binary n-cube," *IEEE Trans. Inf. Theory*, vol. 35, no. 5, pp. 976–987, Sep. 1989.
[11] C.-K. Wen, S. Jin, K.-K. Wong, J.-C. Chen, and P. Ting, "Channel estimation for massive MIMO using Gaussian-mixture Bayesian learning," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1356–1368, Mar. 2015.
[12] J. Moon and Y. Lim, "A reinforcement learning approach to access management in wireless cellular networks," *Wireless Commun. Mobile Comput.*, Vol. 2017, pp. 1–7, May 2017, Art. no. 6474768, doi: 10.1155/2017/6474768.
[13] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Commun.*, vol. 24, no. 2, pp. 98–105, Apr. 2017, doi: 10.1109/MWC.2016.1500356WC.
[14] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 1998.
[15] R. Ali, N. Shahin, Y.-T. Kim, B.-S. Kim, and S. W. Kim, "Channel observation-based scaled backoff mechanism for high-efficiency WLANs," *Electron. Lett.*, vol. 54, no. 10, pp. 663–665, May 2018.
[16] E. Alpaydm, *Introduction to Machine Learning*, 3rd ed. Cambridge, MA, USA: MIT Press, 2014.
[17] G. Alnwaimi, S. Vahid, and K. Moessner, "Dynamic heterogeneous learning games for opportunistic access in LTE-based macro/femtocell deployments," *IEEE Trans. Wireless Commun.*, vol. 14, no. 4, pp. 2294–2308, Apr. 2015.
[18] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer system," Eastern Res. Lab., Digit. Equip. Corp., Maynard, MA, USA, DEC Res. Rep. TR-301, 1984.
[19] *The Network Simulator ns-3*. Accessed: Mar. 5, 2018. [Online]. Available: https://www.nsnam.org/

RASHID ALI received the B.Sc. degree in information technology from Gomal University, Pakistan, in 2007, the M.Sc. degree in computer science (advanced network design), in 2010, under the supervision of Dr. S. Belenki, and the M.Sc. degree in informatics from University West, Sweden, in 2013, under the supervision of Dr. M. Spante. He is currently pursuing the Ph.D. degree with the Wireless Information Networking Laboratory, Department of Information and Communication Engineering, Yeungnam University, South Korea. Between 2007 and 2009, he was a WiMAX Engineer with the Operations Research Department, Wateen Telecom Pvt. Ltd., Pakistan. From 2013 to 2014, he was a Lecturer with COMSATS University, Vehari, Pakistan. His research interests include enhancement of efficiency and reliability in future WLANs, modeling and analyzing the stochastic process of media access control layer resource allocation in future WLANs, deep learning, blockchain, and the Internet of Things.

NURULLAH SHAHIN received the B.Sc. and M.Sc. degrees from the Department of Information and Communication Engineering, Islamic University, Kushtia, Bangladesh, in 2009 and 2010, respectively. He is currently pursuing the combined M.Sc. and Ph.D. degrees with the Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, South Korea. He is a Maintenance Engineer with the IT Operation and Communication Department, Bangladesh Bank (Central Bank of Bangladesh), Bangladesh. His research interests include dense wireless networks, vehicular ad-hoc networks, and resource allocation in wireless networks.

YOUSAF BIN ZIKRIA (SM'17) received the B.Sc. degree in computer engineering from the University of Arid Agriculture Rawalpindi, in 2005, the M.Sc. degree in computer engineering from Comsats University Islamabad, Pakistan, in 2007, and the Ph.D. degree from the Department of Information and Communication Engineering, Yeungnam University, South Korea, in 2016. From 2007 to 2011, he was a Research Officer with Horizon Technology Pvt. Ltd., Pakistan. From 2011 to 2012, he was a Lecturer with King Khalid University, Saudi Arabia. From 2016 to 2018, he was a Postdoctoral Fellow with the Department of Information and Communication Engineering, Yeungnam University, where he is currently an Assistant Professor. He has more than 10 years of experience in research, academia, and industry in the field of information and communication engineering, and computer science. His research interests include the Internet of Things, 5G, wireless communications and networks, opportunistic communications, wireless sensor networks, routing protocols, cognitive radio ad hoc networks, cognitive radio ad hoc sensor networks, transport protocols, VANETS, embedded systems, and information security.

BYUNG-SEO KIM (M'02–SM'17) received the B.S. degree in electrical engineering from In-Ha University, Incheon, South Korea, in 1998, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Florida, in 2001 and 2004, respectively. His Ph.D. research was supervised by Dr. Y. Fang. Between 1997 and 1999, he was a Computer Integrated Manufacturing Engineer of advanced technology research and development with Motorola Korea Ltd., Paju, South Korea. From 2005 to 2007, he was a Senior Software Engineer of networks and enterprises with Motorola Inc., Schaumburg, IL, USA, where he focuses on designing protocol and network architecture of wireless broadband mission critical communications. From 2012 to 2014, he was the Chairman of the Department of Software and Communications Engineering, Hongik University, South Korea, where he is currently a Professor. He has authored or co-authored in around 180 publications and holds 25 patents. His research interests include the design and development of efficient wireless/wired networks including link-adaptable/cross-layer-based protocols, multi-protocol structures, wireless CCNs/NDNs, mobile edge computing, physical layer design for broadband PLC, and resource allocation algorithms for wireless networks. He is a Senior Member of the IEEE and an Associative Editor of the IEEE ACCESS. He served as a member for the Sejong-city Construction Review Committee and the Ansan-city Design Advisory Board. He served as the General Chair for the General Chair of 3rd IWWCN 2017, and a TPC Member for the IEEE VTC 2014-Spring, the EAI FUTURE 2016, and ICGHIT 2016–2019 conferences. He served as a Guest Editor for special issues of the *International Journal of Distributed Sensor Networks*, IEEE ACCESS, *MDPI Sensors*, and *Journal of the Institute of Electrics and Information Engineers*.

SUNG WON KIM received the B.S. and M.S. degrees from the Department of Control and Instrumentation Engineering, Seoul National University, South Korea, in 1990 and 1992, respectively, and the Ph.D. degree from the School of Electrical Engineering and Computer Sciences, Seoul National University, in 2002. From 1992 to 2001, he was a Researcher with the Research and Development Center, LG Electronics, South Korea. From 2001 to 2003, he was a Researcher with the Research and Development Center, AL Tech, South Korea. From 2003 to 2005, he was a Postdoctoral Researcher with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, USA. In 2005, he joined the Department of Information and Communication Engineering, Yeungnam University, South Korea, where he is currently a Professor. His research interests include resource management, wireless networks, mobile networks, performance evaluation, and embedded systems.