A Survey on Applications of Model-Free Strategy Learning in Cognitive Wireless Networks

Wenbo Wang *Student Member, IEEE,* Andres Kwasinski *Senior Member, IEEE,* Dusit Niyato *Senior Member, IEEE,* and Zhu Han *Fellow, IEEE*

Abstract—The framework of cognitive wireless radio is expected to endow the wireless devices with the cognition-intelligence ability, with which they can efficiently learn and respond to the dynamic wireless environment. In many practical scenarios, the complexity of network dynamics makes it difficult to determine the network evolution model in advance. As a result, the wireless decision-making entities may face a black-box network control problem and the model-based network management mechanisms will be no longer applicable. In contrast, model-free learning has been considered as an efficient tool for designing control mechanisms when the model of the system environment or the interaction between the decision-making entities is not available as a-priori knowledge. With model-free learning, the decision-making entities adapt their behaviors based on the reinforcement from their interaction with the environment and are able to (implicitly) build the understanding of the system through trial-and-error mechanisms. Such characteristics of model-free learning is highly in accordance with the requirement of cognition-based intelligence for devices in cognitive wireless networks. Recently, model-free learning has been considered as one key implementation approach to adaptive, self-organized network control in cognitive wireless networks. In this paper, we provide a comprehensive survey on the applications of the state-of-the-art model-free learning mechanisms in cognitive wireless networks. According to the system models that those applications are based on, a systematic overview of the learning algorithms in the domains of single-agent system, multi-agent systems and multi-player games is provided. Furthermore, the applications of model-free learning to various problems in cognitive wireless networks are discussed with the focus on how the learning mechanisms help to provide the solutions to these problems and improve the network performance over the existing model-based, non-adaptive methods. Finally, a broad spectrum of challenges and open issues is discussed to offer a guideline for the future research directions.

Index Terms—Cognitive radio, heterogeneous networks, decision-making, reinforcement learning, game theory, model-free learning.

I. Introduction

A. Cognitive Radio Networks

The original concept of Cognitive Radio (CR) was first proposed a little over one decade ago [1]. In a broad sense, CR is defined as a prototypical radio framework that adopts a radio-knowledge-representation language for the software-defined radio devices to autonomously learn about the dynamics of radio environments and adapt to changes of application/protocol requirements. In recent years, Cognitive Radio Networks (CRNs) have been widely recognized from a high-level perspective as *an intelligent wireless communication system*. A device in a CRN is expected to be aware of its surrounding environment and uses the methodology of understanding-by-building to reconfigure the operational parameters in real-time, in order to achieve the optimal network performance [2], [3]. In the framework of CRNs, the following abilities are typically emphasized:

- radio-environment awareness by sensing (cognition) in a time-varying radio environment;
- autonomous, adaptive reconfigurability by learning (intelligence);
- cost-efficient and scalable network configuration.

Many recent studies on CR technologies focus on radio-environment awareness in order to enhance spectrum efficiency. This leads to the concept of Dynamic Spectrum Access (DSA) networks [4], which are featured by a novel PHY-MAC architecture (namely, primary users vs. secondary users) for opportunistic spectrum access based on the detection of spectrum holes [5]. It is worth noting that by emphasizing the network architecture of spectrum sharing between the licensed/primary networks and the unlicensed/secondary networks [4], "DSA networks" is frequently considered a terminology that is interchangeable with "CR networks" [3]. The rationale behind such a consideration is that a secondary network relies on spectrum cognition modules to make proper decisions for seamless spectrum access without interfering the primary transmissions. For this category of works in the literature, "learning" is mostly about the techniques of feature classification for primary signal identification [6]. For an overview of the relevant techniques, the readers may refer to recent survey works in [7]–[9].

However, in order to achieve autonomous and cost-efficient network configuration, the functionalities of self-organized, adaptive reconfigurability also become fundamental for CRNs, since these functionalities shape the mechanisms of network control and transmission strategy acquisition. By emphasizing such an objective, the network management mechanism is required to dynamically characterize the situation of the decision-making entities in the network and accordingly infer the proper transmission strategies. As the network management mechanisms in conventional wireless networks are ac-

Wenbo Wang and Andres Kwasinski are with the Department of Computer Engineering, Rochester Institute of Technology, Rochester, NY 14623 USA (email: wxw4213@rit.edu, axkeec@rit.edu).

Dusit Niyato is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (email: dniyato@ntu.edu.sg).

Zhu Han is with the Department of Electrical and Computer Engineering as well as the Department of Computer Science, University of Houston, TX 77004 USA (email: zhan2@uh.edu).
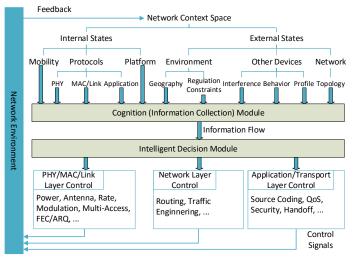
Fig. 1. Relationship between the functionalities of cognition and intelligence in a cognitive wireless network.

quiring more and more levels of such a cognition-intelligence ability, the border between a pure CRN (namely, a CRN in the sense of DSA networks) and a conventional wireless network is gradually diminishing [10], [11]. In recent years, the emerging networking technologies (e.g., CRNs and self-organized networks [12], [13]) emphasize more on autonomous, adaptive reconfigurability. For these networks, the concept of "intelligent network management" based on "cognition" can be re-defined as providing the functionalities of autonomous transmission policy adaptation according to the radio-environment awareness capability of the CR devices in numerous dimensions across the networking protocol stacks [11]. In Figure 1, we provide an overview of the perceivable network states for cognition and the cross-layer network functionalities for configuration in cognitive wireless networks. Interested readers are referred to recent surveys such as [14], [15] for more details about the CR applications in different protocol layers.

Considering the distributed nature of wireless networks, a good CR-based framework of autonomous network configuration in time-varying environments needs to address the following questions:

1) How to properly configure the transmission parameters with limited ability of network modeling or environment observation?
2) How to coordinate the distributed transmitting entities (e.g., end users and base stations) with limited resources for information exchange?
3) How to guarantee the network convergence under the condition of interest conflicts among transmitting entities?

The need to address question 1) lies in the fact that in practical scenarios, the abilities of environment perception may be limited on different levels and/or for different devices. Therefore, the solution to the problems raised by question 1) requires that a decision making mechanism should be able to learn the transmission policies without explicitly knowing the accurate mathematical model of the networks beforehand. Meanwhile, questions 2) and 3) are raised by the basic requirement of a self-organized, distributed control system. Only by

addressing questions 2) and 3) can the network configuration process be efficient in both information acquisition and policy computation. In summary, the key to answering questions 1), 2) and 3) lies in the prospect of enabling the devices in CRNs to distributively achieve their stable operation point under the condition of information incompleteness/locality.

B. From Model-Based Network Management to Model-Free Strategy Learning

When the designer of (distributed) network-controlling mechanisms has complete and global information, the network control problem are frequently addressed in the model-base ways such as the optimization-decomposition-based formulation/solution [16]. With a model-base design methodology, the network control algorithms are usually designed as a set of distributed computations by the network entities (also known as decision-making agents in the domain of control theory) to solve a global constrained optimization problem through decomposition. Under such a framework, since the model of the network dynamics is known in advance, there is no need for "learning" anything about the network dynamics other than the time-varying network parameters. However, in order to adopt such a design methodology, it is necessary to assume that the set of the network parameters (e.g., channel information and channel availability probabilities) that determines the target network utilities is fully available or perfectly known to all the CR devices[1]. If an equilibrium [18] of a multi-entity network is expected instead of the global optimality, the game theoretic approaches (e.g., for multiple access problems [19] and network security problems [20]) can also be adopted. Similar to the optimization-decomposition-based solutions, the game theoretic approaches may still depend on a pre-known model of the network dynamics. In this case, the mathematical tools of optimization theory can also be used for the game theoretic approaches to achieve the goal of obtaining an equilibrium or locally optimal payoff, given that the strategies of the other network entities are accessible.

However, due to the practical limitation of information incompleteness/locality, directly applying the model-based solutions will face difficulties since a model of the network dynamics may even not be available in advance, or in most cases its details may be inaccurate or not instantaneously known to every device. Under the model-based framework, the attempts to conquer the obstacles of information incompleteness/inaccuracy are limited within a small scope by allowing more uncertainty/inaccuracy in the a-priori network model. Examples of these attempts include the introduction of robust control (e.g., variation inequality for spectrum sharing [21]) and fuzzy logic (e.g., fuzzy logic for call admission control [22]). Nevertheless, these techniques still lack the strength of fully addressing the three questions raised in Section I-A.

The difficulty of obtaining an accurate model in advance for dynamic network control in practical scenarios can be illustrated by a multimedia transmission task over an one-hop OFDM-based ad-hoc network (Figure 2). In the network

---

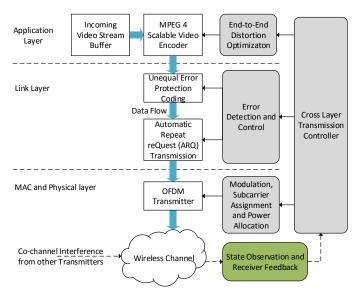[1]More details about the common assumptions for the model-based methods can be found in [17].

Fig. 2. Scalable video transmission over a one-hop OFDM-based ad-hoc network.

Fig. 3. Cognition cycle of a single wireless device [1].

illustrated by Figure 2, the goal of the transmitter-receiver pairs is to achieve the minimized end-to-end distortion through joint power allocation, channel code adaptation and source coding control over dynamic channels. In the practical situation, the obstacle for obtaining an appropriate device behavior model first lies in the difficulty in constructing an accurate end-to-end rate-distortion model at the source codec level, since modeling the rate-distortion relationship for MPEG-4 Scalable Video Coding (SVC) mechanism is notoriously difficult [23]. Moreover, one analytical model may only apply to a certain category of video sources [23]. Meanwhile, the stochastic evolution of the channel condition makes it difficult to predict the transition of the states for the channel-coding/retransmission mechanism, which in return will result in uncertain error propagation at the video decoder of the receiver [24]. Furthermore, when distributed power control and subcarrier allocation mechanism is adopted, it is impractical for a transmitter-receiver pair to fully observe the transmission behaviors of the other pair of nodes, thus rendering the optimal power-channel allocation difficult with merely the local channel observation. As a result, without knowing the end-to-end distortion model, the channel evolution model and the information of peer-node behaviors, the wireless nodes are facing a black-box optimization problem with a limited level of coordination. In this situation, it will be difficult to apply the aforementioned model-based methods for the solution of video transmission control.

In the scenarios of black-box network optimization/control with limited signaling, it is highly desirable that the network control mechanisms do not depend on the a-priori design of the devices' behavior model. As a result, the methods of controlling-by-learning without the need for the a-priori network model, namely, the model-free decision-making approaches [25], [26], are considered more proper, especially within the framework of CR technologies. In the context of adaptive control, controlling-by-learning in CRNs is usually described by the cognition-decision paradigm (Figure 3) [1]. This paradigm describes the learning-based strategy-taking process of a single device from a high-level perspective and
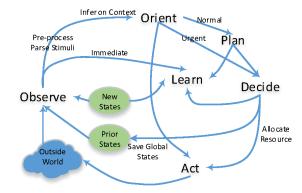
interprets it as a cognition cycle to present the information flow from environment cognition to the final network control decision. In the paradigm, the model-based decision making process is replaced by the observation-decision-action-learning loop. However, the paradigm itself does not provide any detail on how much information about the system model should be learned before a proper transmission strategy can be determined, or in what way the information could be learned.

Under the settings of not knowing a network model in advance, the strategy-learning process can be further divided into two categories according to the ways of using the model knowledge obtained from the learning process: the "model-dependent" methods and the "model-free" methods [25]. For model-dependent learning, an arbitrary division exists between the learning phase and the decision phase, and the goal of learning is to construct the network model first and then use it to derive the network control strategies. By contrast, model-free learning directly learns the network controller without explicitly learning the network model in advance. Early research has pointed out that the model-dependent learning methods are generally more computationally intensive, while model-free learning makes a trade-off of the time to reach controller convergence for reducing computational complexity [25]. Although most of the existing research on strategy-learning methods in wireless networks focus on model-free learning due to the limited computational resources in mobile devices, recent years have seen a tendency that the border between the two categories of strategy-learning methods keeps diminishing [26].

C. A Brief Review of the Existing Survey Works on Learning in CRNs

As indicated by our discussion in Sections I-A and I-B, the problem domain of learning in cognitive wireless networks can be divided into two categories: the problems of wireless environment cognition (namely, spectrum sensing) [7]–[9] and the problems of network management (namely, strategy learning). The solutions to the former problem sub-domain generally provide the information that works as the feed-in to the strategy managers of the latter problem sub-domain. In the literature, the existing surveys on the network management problems are generally organized in accordance with the protocol layers of the OSI/ISO model. These problems include the DSA-based MAC protocol design in CRNs [3], [4], [27], [28], routing protocol design in CRNs [14], [29] and cross-layer

TABLE I
SUMMARY OF EXISTING SURVEY WORKS ON CR NETWORKING PROBLEMS AND MODEL-FREE LEARNING METHODS

Problem Domain of Cognitive Wireless Networks	Sub-domain of CR Networking Problems	Category of Corresponding Machine-learning Methods	Sub-category of Learning Methods
Wireless environment cognition	Spectrum sensing [7]–[9]	Supervised learning (pattern classification) [6], [32]	N/A
Network management	DSA-based MAC protocol design [3], [4], [27], [28]	Unsupervised learning (model-free learning) [25], [26], [33]–[35]	Single-agent-based reinforcement learning [25]
	Spectrum-aware routing [14], [29]		Multi-agent-based reinforcement learning [26], [34]
	Self-organization [12], [30]		Learning automata [35]
	Network security [20], [31]		Repeated-game-based learning [33], [36], [37]

network control problems in CRNs such as self-organization [12], [30] and network security problems [20], [31].

With respect to different domains of networking problems, the pool of the potential machine-learning-based solutions can also be grouped into two major categories. For the problems of spectrum sensing, the survey on applications of signal-classification-oriented learning methods can be found in recent studies such as [6], [32]. For the network management problems, the (model-free) strategy-learning-based solutions are generally identified as belonging to the category of un-supervised learning [6]. More specifically, the techniques of controlling-by-learning in CRNs are usually featured by the trial-and-error interactions with the dynamic wireless environment and thus also known as "reinforcement learning" (see our discussion in Section II). In the past decade, researchers have paid a significant attention to the confluence of adaptive control, model-free learning and game theory [26], [33]. In the domain of CRNs, it is believed that such a trend will lead to a promising solution of the various network control/resource allocation problems (e.g., [28], [31]). In return, the development of the recent network technologies, such as self-organized networks and CRNs, is increasingly demanding more efficient learning mechanisms to be implemented for an adaptive, self-organized solution.

Most of the existing model-free learning methods for network control in CRNs find their origin in the domain of control theory. In the literature, important surveys on these model-free learning methods from the perspective of control/game theory include [25], [26], [33]–[35]. In the context of network control, existing survey works on the applications of strategy learning usually focus on a certain sub-category of these learning methods. In [36], [37], comprehensive surveys on distributed learning mechanisms are provided based on the framework of repeated games (see our discussion in Section II-C). In [6], [38], the surveys on model-free learning in CRNs place the focus more directly on the Q-learning based methods (see our discussion in Section II-A). Apart from the aforementioned works, other survey works on strategy learning in wireless networks usually focus on a specific sub-domain of applications such as wireless ad-hoc networks [39] and sensor networks [40]. To assist the readers in obtaining an overview of the development of model-free learning methods and their relationship with the network management problems in CRNs, we summarize the aforementioned survey works according to the domains they belong to in Table I.

TABLE II
SUMMARY OF ACRONYMS FOR WIRELESS NETWORKING TERMINOLOGIES

Terminologies	Abbreviations
Base station	BS (Section III, V)
Cognitive radio	CR (Section I, III, IV, V)
Cognitive radio networks	CRNs (Section I, III, IV, V)
Dynamic channel assignment	DCA (Section III)
Dynamic spectrum access	DSA (Section I, III)
Key performance indicator	KPI (Section VI)
Network operator	NO (Section V)
Primary user	PU (Section III, IV, V)
Signal-to-interference-plus-noise-ratio	SINR (Section IV, V)
Signal-to-noise-ratio	SNR (Section III, V)
Service provider	SP (Section V)
Secondary user	SU (Section III, IV, V)
Heterogeneous networks	HETNET (Section IV)

D. Organization of the Paper

This paper is devoted to providing a comprehensive survey on the current development of model-free learning in the context of the cognitive wireless networks. In order to highlight the difference in the existing level of information incompleteness/locality (from another perspective, the degree of information coupling) for different learning mechanisms, we organize the survey on the applications of learning in CRNs into three major categories: (a) strategy learning based on the single-agent systems, (b) strategy learning based on the loosely coupled multi-agent systems and (c) strategy learning in the context of games. In Section II, the necessary background and the preliminary concepts of learning in the single-agent system, the distributed, multi-agent systems and games are provided. In Section III-V, the recent research on the applications of the three major categories of model-free learning mechanisms in CRNs is reviewed according to the different system models that the learning mechanisms are based on. In Section VI, some important open issues for the application of model-free learning in CRNs are outlined in order to provide the insight into the future research directions. Finally, we summarize and conclude the paper in Section VII. In Table II and Table III, we provide an acronym glossary of the terms used in the paper.

II. BACKGROUND: MODEL-FREE LEARNING IN THE DOMAINS OF DISTRIBUTED CONTROL AND GAME THEORY

Although the applications of model-free learning in wireless networks only became more commonplace in the early 2000s, the fundamental development of the model-free learning theory can be traced back much earlier, to the 1980s [41], [42].

TABLE IV
SEQUENTIAL DECISION-MAKING MODELS IN A NUTSHELL

General Model	Specific Model	Tuple-Based Model Description	Agent-Strategy Coupling	Objective	Utility Measurement
Multi-agent Markov Decision Process (MDP) / Stochastic Game (SG)	Single-agent MDP	$\langle \mathcal{S}, \mathcal{A}, r, \Pr(\mathbf{s}'\|\mathbf{s}, a) \rangle$	N/A	Utility optimization	Accumulated utility
	Multi-agent MDP	$\langle \mathcal{N}, \mathcal{S} = \times \mathcal{S}_n, \mathcal{A} = \times \mathcal{A}_n, \{r_n\}_{n \in \mathcal{N}}, \Pr(\mathbf{s}'\|\mathbf{s}, \mathbf{a}) \rangle$	Allowed	Utility optimization	Accumulated utility
	Stochastic games	$\langle \mathcal{N}, \mathcal{S} = \times \mathcal{S}_n, \mathcal{A} = \times \mathcal{A}_n, \{r_n\}_{n \in \mathcal{N}}, \Pr(\mathbf{s}'\|\mathbf{s}, \mathbf{a}) \rangle$	Always	Reaching equilibria	Accumulated utility
	Repeated games	$\langle \mathcal{N}, \mathcal{A} = \times \mathcal{A}_n, \{r_n\}_{n \in \mathcal{N}} \rangle$	Always	Reaching equilibria	Accumulated utility
	Static games	$\langle \mathcal{N}, \mathcal{A} = \times \mathcal{A}_n, \{r_n\}_{n \in \mathcal{N}} \rangle$	Always	Reaching equilibria	Instantaneous utility

TABLE III
SUMMARY OF ACRONYMS FOR MODEL-FREE LEARNING TERMINOLOGIES

Terminologies	Abbreviations
Actor-critic learning	AC-learning (Section II, VI)
Actor-critic learning automata	ACLA (Section II)
Correlated equilibrium	CE (Section II, V)
Correlated-Q learning	CE-Q learning (Section V)
Constrained Markov decision process	CMDP (Section III)
COmbined fully DIstributed PAyoff and Strategy-Reinforcement Learning	CODIPAS-RL (Section II, V)
Derivative-action gradient play	DAGP (Section V)
Dynamic programming	DP (Section II)
Distributed reward and value function	DRV function IV
Distributed value function	DVF (Section IV)
Experience-weighted attraction learning	EWAL (Section VI)
Fictitious play	FP (Section II, V)
Greedy policy searching in the limit of infinite exploration	GLIE (Section II)
Gradient play	GP (Section III)
Learning automata	LA (Section II, V)
Linear-reard-inaction algorithm	L_{R-I} (Section II, V)
Multi-agent Markov decision process	MAMDP (Section II, III)
Multi-agent system	MAS (Section II, IV, V, VI)
Markov decision process	MDP (Section II, III)
Nash equilibrium	NE (Section II, V, VI)
Observation-orient-decision-action loop	OODA loop (Section I)
Partially observable Markov decision process	POMDP (Section III, IV)
Single-agent Markov decision process	SAMDP (Section II, III)
State-action-reward-state-action	SARSA (Section II, III)
Single-agent system	SAS (Section II, III)
Stochastic games	SGs (Section II, V)
Smoothed/Stochastic fictitious play	SFP (Section V)
Simulated perturbation stochastic approximation	SPSA (Section V)
Temporal difference learning	TD-learning (Section II, III, VI)
Transfer learning	TL (Section VI)

TABLE V
SUMMARY OF THE MAIN NOTATIONS IN SECTION II

Symbol	Meaning
t	Timing index
a	A single action of the decision-making agent in a single-agent system
a_{-n}	The joint action of the adversary agents for agent n in a game
$\mathcal{A}_n$	A finite set of actions for agent n in a multi-agent system
s	A single environment state of the agent in a single-agent system
$\mathcal{S}_n$	A finite set of environment states for agent n in a multi-agent system
$u_n(s_n, a_n)$ or u_n	Instantaneous utility function of agent n in a multi-agent system
$\Pr(\cdot)$	State transition probability function
β	The discount factor for a discounted-reward MDP
$\pi(s, a)$ or π	The policy mapping function of an agent from a given state to an action
π^*	An optimal or equilibrium policy
$\pi(s, a_{-n})$ or π_{-n}	The joint policy of the adversary agents for agent n in a game
$V_\beta^\pi(s)$	The state-value function of a discounted-reward MDP from the starting state s
$Q_\beta^\pi(s, a)$	The state-action value function of a discounted-reward MDP from taking action a at the starting state s
$h^\pi(s)$	The state-value function of an average-reward MDP from the starting state s
$V^\pi(s)$	The bias utility of an average-reward MDP from taking policy π at starting state s
α_t, θ_t	The learning rates
$\tilde{r}$	The (normalized) value of environment response used by learning automata algorithms

this section are list in Table V.

A. Single-Agent Strategy Learning

In this section, we provide a necessary introduction of the general-purpose learning methods that are developed in the domains of distributed control and game theory. To assist our discussion about learning techniques applied to cognitive wireless networks, we categorize the learning methods by the degree of coupling among the decision-making agents with respect to different system models. In what follows, we will briefly introduce the general-purpose learning algorithms that are built upon the decision-making models of single-agent systems, loosely coupled multi-agent systems and game-based multi-agent systems. Before proceeding to more details of the learning mechanisms, we first provide an overview of these decision-making models in Table IV. The notations used in

In the context of distributed control and robotics, single-agent learning has been considered as the most fundamental class of the strategy-learning methods. Single-agent learning generally assumes that the learning agent has full access to the state information that can be obtained about the system. Frequently, the terminologies "reinforcement learning" and "model-free learning" are (partially) used interchangeably to refer to the decision-making process of a single agent. The agent learns to improve its performance by merely observing the state changes in its operational environment and the utility feedback that it received after taking an action. In the recent surveys on reinforcement-learning theory and its applications [6], [38], such a decision-learning process is
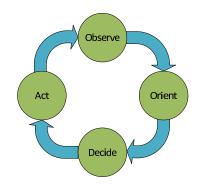
Fig. 4. The OODA loop, often known as the cognition cycle [1].

described by an abstract model, namely, the Observe-Orient-Decision-Action (OODA) loop [4]. The OODA loop (Figure 4) can be considered as a generalized model of the cognition cycle in the context of cognitive wireless networks (Figure 3), and it provides a generic description of the information flow in the intelligent decision-making process. However, it is the task of the specific reinforcement-learning methods to define the rules of agent behaviors that guide the interaction with the to-be-explored environment. Since in most of the practical scenarios, a learning agent needs to deal with environment uncertainty, in the literature, a Markov Decision Process (MDP) [43] becomes a prevalent tool for abstracting the model of the agent-environment interaction. Based on the MDP framework, various model-free learning methods such as Temporal Difference (TD) learning [44] and learning automata [35] can be adopted to define the behavior rules of an agent.

The standard (single-agent) MDP model is used to describe a stochastic Single-Agent System (SAS). Mathematically, a single-agent MDP is defined as follows:

Definition 1 (Single-agent MDP [26]). *A single-agent MDP is defined as a 4-tuple: $\langle \mathcal{S}, \mathcal{A}, u, \Pr(\mathbf{s}'|\mathbf{s}, a)\rangle$, in which*

- $\mathcal{S} = \{s_1, \ldots, s_{|\mathcal{S}|}\}$ *is a finite set of environment states,*
- $\mathcal{A} = \{a_1, \ldots, a_{|\mathcal{A}|}\}$ *is a finite set of agent's actions,*
- $u : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ *is the instantaneous utility function,*
- $\Pr : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ *is the state transition probability function, which retains the Markovian property.*

In the MDPs, the underlying environment is a stationary stochastic process, and the consequences of the decisions can be probabilistic. The goal of a decision-learning agent is to find the proper stationary policy, $\pi = \Pr(a|s)$ that probabilistically maps state s to action a so that the accumulated long-term utility of the agent is optimized. With respect to different applications, the objectives of the MDPs may appear in different forms. In this survey, we will mainly consider two types of the infinite-horizon objectives [25] as follows:

- the discounted-reward MDP with the discount factor $\beta \in [0, 1]$:

$$V_\beta^\pi(s) = E_\pi \left(\sum_{t=0}^{\infty} \beta^t u_t(s_t, a) \right), \qquad (1)$$

- the average-reward MDP:

$$h^\pi(s) = \lim_{T \to \infty} \frac{1}{T} E_\pi \left(\sum_{t=0}^{T-1} u_t(s_t, a) \right). \qquad (2)$$

Both types of MDPs can be represented in the form of the Bellman optimality equation. For the discounted-reward MDP, the Bellman equation can be represented either by the state-value function starting from state s under policy π:

$$V_\beta^\pi(s) = E_\pi(u(s, a)) + \sum_{s' \in \mathcal{S}} \Pr(s'|s, \pi) V_\beta^\pi(s'), \qquad (3)$$

or by the state-action value function (Q-function) that starts from taking action a at state s and follows policy π thereafter:

$$Q_\beta^\pi(s, a) = u(s, a) + \sum_{s' \in \mathcal{S}} \Pr(s'|s, a) V_\beta^\pi(s'). \qquad (4)$$

In order to express the average-reward MDP in the form of the Bellman equation, the average adjusted sum of utility (i.e., bias) following policy π is introduced as follows:

$$V^\pi(s) = \lim_{T \to \infty} E_\pi \left(\sum_{t=0}^{T-1} (u_t(s_t, a) - h^\pi(s)) \right), \qquad (5)$$

with which the average-reward MDP can be expressed by the state-value function[2]:

$$V^\pi(s) + h^\pi(s) = E_\pi(u(s, a)) + \sum_{s' \in \mathcal{S}} \Pr(s'|s, \pi) V^\pi(s'). \qquad (6)$$

With a variety of on-line learning methods that estimate the optimal Q-value or the bias value, a broad spectrum of value-iteration-based learning algorithms have been proposed [26], [45]. Among them, the most widely used model-free learning algorithm is Q-learning [44], which estimates the state-action value in (4) of a discounted MDP based on the time difference of the estimated values for the state-action value function:

$$Q_{t+1}(s_t, a_t) \leftarrow Q_t(s_t, a_t) + \alpha_t \bigg(u_t(s_t, a_t) \\ + \beta \max_{a'} Q_t(s_{t+1}, a') - Q_t(s_t, a_t) \bigg), \qquad (7)$$

where $\alpha_t \in (0, 1]$ is the learning rate specifying the step that the current state-action value is adjusted toward the TD sample $u(s_t, a_t) + \beta \max_{a'} Q_k(s_{t+1}, a')$. Q-learning in (7) has been proved to be able to converge to the true optimal value of the state-action value function with a stationary deterministic policy, given that $\sum_{t=0}^{\infty} \alpha_t = \infty$, $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$ and all actions in all states are visited with a non-zero probability [44]. The model-free property of Q-learning is reflected in the iterative approximation procedure for the Q-values, which does not require knowing the transition map $\Pr(s'|s, a)$ of the MDP in advance.

The counterpart to Q-learning in the average-reward MDP is known as R-learning [45]. In addition to learning the state-action value of the bias expressed in (5), R-learning also needs to learn the estimate of the average reward h^π. Therefore, R-learning is performed by a two-time scale learning process:

$$R_{t+1}(s_t, a_t) \leftarrow R_t(s_t, a_t) + \alpha_t \big(u_t(s_t, a_t) + \max_{a'} R_t(s_{t+1}, a') \\ - h_t - R(s_t, a_t) \big), \qquad (8)$$

---

[2]Due to the space limit, the conditions for the existence of a value function in the form of (6) is not presented here. The readers are referred to [45] for the details.

$$h_{t+1} \leftarrow h_t + \theta_t \big(u(s_t, a_t) + \max_{a'} R_t(s_{t+1}, a') - h_t \\ - \max_{a'} R_t(s_t, a') \big). \qquad (9)$$

In contrast to the value-iteration-based learning algorithms given in (7), (8) and (9), the decision-learning methods based on the Learning Automata (LA) allow an agent to directly learn the stationary randomized policy. Instead of updating the action according to the myopic optimal Q-value in discounted-reward MDP and bias-value in average-reward MDP, the LA directly updates the probabilities of actions based on the utility feedback [35]. Let the action probability vector at time instance t be $\boldsymbol{\pi}(t) = (\pi_1(t), \ldots, \pi_{|\mathcal{A}|}(t))$, where $|\mathcal{A}|$ is the size of the action set. Then an LA-based algorithm should be able to achieve the following goal [35]:

$$\boldsymbol{\pi}^* = \max_{\pi(t)} E[\tilde{r}(t) | \boldsymbol{\pi}(t), s(t)], \qquad (10)$$

where $\tilde{r}$ is the value of environment response, and is usually generated based on the instantaneous reward u_t as a normalized value (i.e., $\tilde{r} \in \{0, 1\}$). The general updating rule for LA can be expressed as follows [46]:

$$\begin{cases} \pi_i(t{+}1) = \pi_i(t) - (1{-}\tilde{r}(t)) f_i(\pi_i(t)) + \tilde{r}(t) g_i(\pi_i(t)), \\ \qquad\qquad\qquad\qquad\qquad\qquad \forall a(t) \neq a_i, \\ \pi_i(t{+}1) = \pi_i(t) + (1{-}\tilde{r}(t)) \sum_{j \neq i} f_i(\pi_i(t)) - \\ \qquad \tilde{r}(t) \sum_{j \neq i} g_i(\pi_i(t)), \qquad a(t) = a_i, \end{cases} \qquad (11)$$

where f and g are the penalty and reward functions, respectively. Specifically, different forms of f and g lead to different learning schemes. Among them, it has been proved that the linear-reward-inaction (i.e., L_{R-I}) algorithm is guaranteed to achieve the ϵ-optimal policies [47]. In [45], the automaton-updating procedure based on L_{R-I} is adopted to learn the optimal policy in the ergodic MDPs with average-reward objectives[3]. In other works such as [48], the optimal policy of the discounted-reward MDP is learned by adopting the L_{R-I} algorithm for policy updating and the standard Q-learning algorithm in (7) for Q-value estimation at the same time.

Although the two groups of learning mechanisms, namely, value-iteration-based learning (e.g., TD-based learning such as Q-learning and R-learning) and LA-based learning appear distinct from each other, both of them can be considered as special cases in the framework of Actor-Critic (AC) learning [49]. In the context of AC learning, the concepts of value function and policy are also known as "critic" and "actor", respectively. Since Q-learning and R-learning only learn a state-action value function and there is no explicit function for the policy, the two learning algorithms are also known as the critic-only algorithms. On the contrary, without using any form of a stored value function, LA can be considered an actor-only algorithm. Extending from these two special cases, a generalized AC-based mechanism keeps track of both the state-value function and the policy evolution at the same time. In this sense, a generalized AC-based mechanism is also known as combined payoff and strategy learning [37]. Specifically, if the state-action value of the MDP is learned following the TD-based methods and in the meanwhile the
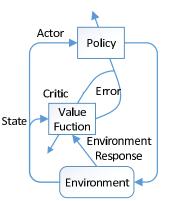

Fig. 5. Schematic view of the generalized AC algorithm.

learning agent's policy is updated following the LA-based methods, the AC-learning mechanism is also known as Actor-Critic LA (ACLA) [50]. A typical rule for jointly updating the estimate of the state-value and policy in ACLA can be found in [50]. Here it is worth noting that for both critic and actor updating, the learning mechanisms are not limited to the aforementioned two categories of algorithms. For example, an on-policy learning algorithm, i.e., State-Action-Reward-State-Action (SARSA)[4], can be used to replace the Q-learning-based critic-updating mechanism, and instead of the LA-like actor-updating mechanism, policy gradient is widely used for actor updating [49]. A schematic overview of the generalized AC algorithm is given in Figure 5.

B. Strategy Learning in the Loosely Coupled Multi-Agent System

A stochastic Multi-Agent System (MAS) can be defined by extending the 4-tuple Single-Agent MDP (SAMDP) (Definition 1) into a 5-tuple Multi-Agent MDP (MAMDP): $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \{u_n\}_{n \in \mathcal{N}}, \Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$, in which $\mathcal{N}$ is the set of the decision-making agents, $\mathcal{S} = \times \mathcal{S}_n$ is the Cartesian product of the local state spaces of all the agents and $\mathcal{A} = \times \mathcal{A}_n$ is the Cartesian product of the local action spaces of all the agents. When considering the learning mechanism in an MAS, it is natural to simply adopt the standard SAS-learning algorithms by assuming that each agent is an independent learner with the local utility function $u_n(s_n, a_n)$. In doing so, the activities of the other agents are treated as part of a stationary environment and the learning agents update their policy without considering their interactions with the other agents. This approach enjoys popularity especially within the studies in the cooperative decision-making domain [52], [53]. Its typical applications can be found in modeling the hunter-prey systems [54] and team coordination [55], just to mention a few. However, it is important to note that multi-agent learning based on SAS learning requires the joint learning process to be decomposed into local ones. Thus, individual-agent behaviors are relatively disjoint, and the agents are able to ignore the information raised by the interactions with each other. This is also the reason for us to call it a "loosely coupled multi-agent system". Otherwise, with concurrent learning, all the individual agents

---

[3]For the details of L_{R-I}, please refer to Section V-A3.

[4]About the difference between Q-learning and SARSA, the readers are referred to [51] for more details.

need to adapt their policies in the dynamic context of the other learners, in which case the basic assumption of stationary environment for the single-agent scenarios will no longer hold.

Although convergence of SAS-based learning is not guaranteed in most of the practical MAS scenarios, attempts of generalizing the convergence condition for the SAS-based learning mechanism can still be found in the literature. By limiting the application scenarios to fully-cooperative MAMDPs (i.e., common-payoff MAMDPs), the convergence property of SAS-based learning with Greedy policy searching in the Limit of Infinite Exploration (GLIE) for MAMDPs is discussed in [56]:

Proposition 1. *For the multi-agent Q-learning schemes obeying the individual updating rule in (7) in a cooperative MAS system, assume that the following conditions are satisfied:*

- *the learning rate α_t decreases over time such that $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$,*
- *each agent samples each of its actions infinitely often,*
- *the probability of agent i choosing action $a \in \mathcal{A}_i$ is nonzero,*
- *the probability of taking a non-optimal action decreases to 0 when $t \to \infty$ during the exploration stage,*

let $\pi_i^*(t)$ be a random variable denoting the probability of action-taking in a (deterministic) equilibrium strategy profile being played at time t. Then for SAS-based learning, for any $\xi, \epsilon > 0$, there exists $T(\xi, \epsilon)$ such that

$$\Pr(|\pi_i^*(t) - 1| < \epsilon) > 1 - \xi, \forall t > T(\xi, \epsilon). \qquad (12)$$

Although lacking a formal mathematical proof, Proposition 1 has been widely accepted in related studies [34], [57]. A more general convergence condition for SAS-based learning in MAS scenarios is given by [58]:

Proposition 2. *In an MAS environment, an agent following the updating rule in (7) will converge to the optimal response Q-function with probability 1 as long as all the other agents converge in behaviors with probability 1. If the agent follows a GLIE policy and its best response policy is unique, it will also converge in behavior with probability 1.*

Propositions 1 and 2 provide theoretical support for the convergence property of a number of SAS-based learning algorithms that can be considered a variation of (7) (e.g., distributed Q-learning in cooperative MAMDPs [59] and policy hill-climbing in two-agent MAMDPs [60]). Again, it is worth pointing out that for most MAS scenarios (e.g., general-sum stochastic games) convergence of SAS-based learning is not guaranteed. Furthermore, even when convergence can be reached, it usually takes a significant amount of time for merely determining switching between one pair of actions. As a results, most of the practical SAS-based learning mechanisms are limited in the special scenarios such as the fully-cooperative MAS or two-agent MAS. In the framework of the independent learning algorithm using standard Q-learning [56], other SAS-based learning algorithms for MAS usually try to eliminate the uncertainty caused by the actions of the other agents while still retaining the distributivity of the decision-making process. One typical example can be found in [59], which projects the global Q-table of a deterministic MAMDP

(namely, the state transition is deterministic in the MDP) using centralized Q-learning with joint action $\mathbf{a} = (a_1, \ldots, a_n)$, $Q(s, \mathbf{a})$, to the local Q-table of agent i with only local action information a_i, $Q(s, a_i)$. Following the standard Q-learning rule, the projection-based independent learning adopts an optimistic assumption that all the other agents will act optimally. However, the learning result of such a distributed algorithm is greedy with respect to the centralized Q-table with the joint action. Additionally, its convergence when extended to the scenarios of stochastic MAMDPs is not guaranteed since it cannot discern the influence of the behaviors of the other agents from that of the state dynamics. It is important to note that without explicit coordination, which is at the cost of losing the distributiveness of the decision-making process, all the independent-learning-based algorithms will suffer for the same reason as in the tightly coupled, MAS-based scenarios.

Despite all the limitations of independent learning, one important benefit of adopting the disjoint learning processes in the MAS is that it creates the opportunities of experience sharing among individual agents. In [54], [61], the "implicit imitation" mechanism by the observer agents is proposed to incorporate the experience of the expert agents in the MAS. Under the framework of distributed, independent MDPs, it is frequently assumed that the learning agents are analogous to each other in terms of state space, state transition and action set [61]. Then experience transferring can be implemented by modifying the estimated state-action value of the observer agent based on the expertise evaluation of the mentor agents and the weighted combination of their respective Q-values [54]. When experience transferring is considered beyond the framework of model-free learning and the model-based policy-learning mechanism is adopted, the observer agent can also implement the experience learning by maintaining the estimation of the mentor's transition map from observation, and incorporating the estimation into its own value-iteration process [61].

C. Multi-Agent Strategy Learning in the Context of Games

In most of the practical scenarios, the dynamics of the multi-agent MDP (e.g., the transition probabilities and the local payoff) is determined by the joint policy of all the agents. To facilitate distributed policy learning, the multi-agent MDP is usually viewed as a Stochastic Game (SG). Mathematically, an SG shares exactly the same 5-tuple structure as an MAMDP, $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \{u_n\}_{n \in \mathcal{N}}, \Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$. However, the goal of each agent in the SG is to maximize its individual payoff [18]. Based on the definition of SGs, a repeated game can be obtained as a 3-tuple, $\langle \mathcal{N}, \mathcal{A} = \times \mathcal{A}_n, \{u_n\}_{n \in \mathcal{N}} \rangle$, by fixing the environment state as invariant while maintaining the objective of each player as maximizing its individual discounted/average payoff over the infinite time horizon. In the repeated game, the system dynamics is reduced to only the mapping between the action and the payoff: $u_n : \mathcal{A} \to \mathbb{R}$. Further, when the repeated game is played only once, it is reduced to a static game. In return, any single shot of an SG or a repeated game is a static game and is known as a single stage or one-shot game of the original game [62].

One important reason for adopting the game theoretic models lies in the requirements that decisions are to be made in a distributed manner with the limited ability of both information acquisition and action coordination. This may be either due to the overwhelming dimension of the state-action space as the number of agents grows, or due to the overhead for information exchange among agents. In the game-based decision-making model, the individual-rationality property of the agents leads to the concept of the best response. In an SG, the best response of agent n is defined as the policy $\{\pi_n = \Pr(\mathbf{s}, a_n) : \mathbf{s} \in \mathcal{S}\}$ such that the long-term payoff under local policy π_n is not worse than that under any other local policies: $V_n(\pi_n, \pi_{-n}) \geq V_n(\pi'_n, \pi_{-n})$, given the joint adversary policy π_{-n}. Here, π_{-n} is the joint strategy of the adversary agents except agent n and V_n can be either the discounted long-term payoff or the average long-term payoff. If $\forall n \in \mathcal{N}$, the policy is a best response to the joint strategy of the other agents, we say that the policy profile $(\pi_1, \ldots, \pi_{|\mathcal{N}|})$ is a Nash Equilibrium (NE) [18]. In the context of games, the goal of policy learning now becomes finding the policy updating rules for reaching a specific equilibrium. Apart from the most commonly used solution concept of NEs, a policy learning mechanism may resort to other types of equilibria for the convenience such as ensuring convergence or improving performance. In order to facilitate our discussion on different learning algorithms, we provide the formal definition of several equilibria in discounted-reward SG $G = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \{u_n\}_{n \in \mathcal{N}}, \Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$ as follows:

Definition 2 (Nash Equilibrium (NE)). *In a game G, an NE point is a tuple of strategies $(\pi_1^*, \ldots, \pi_{|\mathcal{N}|}^*)$ such that $\forall \mathbf{s} \in \mathcal{S}$, $\forall n \in \mathcal{N}$ and $\forall \pi_n \in \Pi_n$,*

$$V_{\beta,n}(\mathbf{s}, \pi_1^*, \ldots, \pi_n^*, \ldots, \pi_{|\mathcal{N}|}^*) \geq V_{\beta,n}(\mathbf{s}, \pi_1^*, \ldots, \pi_n, \ldots, \pi_{|\mathcal{N}|}^*),$$

in which $V_{\beta,n}(\mathbf{s}, \pi_1^*, \ldots, \pi_{|\mathcal{N}|}^*)$ is given by (3) with a slight abuse of notation.

Definition 3 (Correlated Equilibrium (CE)). *In a game G, a CE point is a joint strategy $\pi^* = (\pi_n^*, \pi_{-n}^*)$ such that $\forall n \in \mathcal{N}$, $\forall \mathbf{s} \in \mathcal{S}$ and $\forall a_n, a'_n \in \mathcal{A}_n$,*

$$\sum_{a_{-n} \in \mathcal{A}_{-n}} \pi^*(\mathbf{s}, a_n, a_{-n}) Q_{\beta,i}^{\pi^*}(\mathbf{s}, a_n, a_{-n}) \geq$$
$$\sum_{(a_n, a_{-n}) \in \mathcal{A}} \pi^*(\mathbf{s}, a_n, a_{-n}) Q_{\beta,i}^{\pi^*}(\mathbf{s}, a'_n, a_{-n}),$$

in which $Q_{\beta,i}^{\pi^*}(\mathbf{s}, a_n, a_{-n})$ is given by (4) with a slight abuse of notation and $\pi^*(\mathbf{s}, a_n, a_{-n}) = \pi^*(\mathbf{s}, a_{-n}|a_n)\pi^*(\mathbf{s}, a_n)$.

Definition 4 (ϵ-Equilibrium). *Let $\epsilon > 0$, the profile $\pi^* = (\pi_n^*, \pi_{-n}^*)$ is an ϵ-equilibrium of game G if by following π^* no player can improve its payoff by more than ϵ at any stage.*

Specifically, given the condition of the NE (Definition 2), $\pi^* = (\pi_n^*, \pi_{-n}^*)$ is an ϵ-NE if $\forall \mathbf{s} \in \mathcal{S}$, $\forall n \in \mathcal{N}$ and $\forall \pi_n \in \Pi_n$,

$$V_{\beta,n}(\mathbf{s}, \pi_n^*, \pi_{-n}^*) \geq V_{\beta,n}(\mathbf{s}, \pi_n, \pi_{-n}^*) - \epsilon.$$

Given the condition of CE (Definition 3), $\pi^* = (\pi_n^*, \pi_{-n}^*)$ is an ϵ-CE if $\forall \mathbf{s} \in \mathcal{S}$, $\forall n \in \mathcal{N}$ and $\forall a_n, a'_n \in \mathcal{A}_n$,

$$\sum_{a_{-n} \in \mathcal{A}_{-n}} \pi^*(\mathbf{s}, a_n, a_{-n}) Q_{\beta,i}^{\pi^*}(\mathbf{s}, a_n, a_{-n}) \geq$$
$$\sum_{(a_n, a_{-n}) \in \mathcal{A}} \pi^*(\mathbf{s}, a_n, a_{-n}) Q_{\beta,i}^{\pi^*}(\mathbf{s}, a'_n, a_{-n}) - \epsilon.$$

Based on Definitions 2-4, the conditions of equilibria for repeated/static games can be obtained in a similar way. From the perspective of strategy derivation, a CE can be considered a generalized form of an NE since it does not require the individual player's strategy to be independent with each other. Although the adoption of a CE is recognized as being able to provide a better performance of an NE, such a performance improvement is usually at the cost of introducing an arbitrator or coordinator into the game [18]. From the perspective of convergence reaching, an ϵ-equilibrium can be considered a form of both NE and CE with relaxed condition. For learning algorithm design in repeated games, the introduction of ϵ-equilibrium helps develop the learning mechanisms that guarantee the convergence to near-equilibrium with a limit-inferior bound. However, it is worth noting that for a general SG, the existence of a stationary ϵ-equilibrium is not guaranteed beyond the case of two-player SGs [63].

According to the Folk theorem [36], for every infinite-horizon, n-player, discounted repeated/stochastic game with a finite number of actions, the existence of a stationary policy $\boldsymbol{\pi}^*$ as a subgame-perfect NE [18] is guaranteed. By proving the existence of a subgame-perfect NE, the Folk theorem implies that when compared with the static one-shot game, policy learning may be able to obtain a better payoff with the new NE in the repeated games. Such a benefit is also considered a major motivation for the engineers to adopt the game-based learning algorithms in the domain of distributed decision-making. However, the implementations of the learning algorithms heavily rely on the game structures and the forms of the equilibria, and may differ significantly. Within the past two decades, numerous methods have been proposed for strategy learning in games. In order to facilitate our survey on their applications in cognitive wireless networks, we categorize the model-free learning algorithms along the following dimensions[5]:

1) Value iteration vs. policy iteration: in SGs, most of the learning algorithms based on the state-action value estimation fall into the category of value-iteration based algorithms. These algorithms include minimax Q-learning [64], NSCP-learning [65] Nash Q-learning [66], Nash R-learning [67] and CE-Q learning [68]. In contrary to value-iteration-based learning, the policy-iteration-based learning algorithms directly update the action-probability vectors of each agent, using either the observation of the adversary agents' action pattern or the payoff received from interaction with the environment. These algorithms include standard Fictitious Play (FP) [33], asynchronous best response [69], LA-based learning algorithms (e.g.,

---

[5]All the game-based learning methods to be discussed in the following sections originate from these algorithms, and in Section III more details will be provided for each of them.

L_{R-I} learning [47] and Bush-Mosteller learning [70]), gradient-play-based better reply [71] and no-regret learning [72]. In the cases when both the strategy and the local expected payoff are to be learned, the AC-like, multiple-timescale learning algorithms [73] provide an efficient strategy-learning approach (e.g., stochastic FP [33]) for the agents. Further, when the joint action or the payoff of the adversary agents is not directly observable, conjecture-variation-based learning [74] works as an alternative way of the aforementioned learning algorithms. In the literature, these joint policy-value-iteration mechanisms for games are also known as the COmbined fully DIstributed PAyoff and Strategy-Reinforcement Learning (CODIPAS-RL) mechanisms [37].

2) NE vs. other equilibria: most of the learning algorithms in 1) such as proposed in [47], [64]–[67], [69]–[71] aim at finding the NE of the repeated games/SGs. By contrast, the goal of CE-Q learning [68] and some no-regret learning algorithms [72] is to learn the CE in the SG and the repeated game, respectively. By relaxing the condition of an NE from the profile of real actions to the profile of agent beliefs, conjecture-variation-based learning [74] converges to the conjecture equilibrium [75]. In most practical scenarios based on the framework of general repeated games, FP and stochastic FP only guarantee that the ϵ-equilibrium can be reached [33]. In the literature, ϵ-equilibrium is sometimes known as the Logit equilibrium when the Logit function[6] is used for strategy updating.

3) Noncooperative games, cooperative games and team games: technically, these three major categories cover most of the game-based models in the applications of distributed control. Provided that the noncooperative games satisfy certain properties (e.g., being supermodular/submodular [76] or having a unique NE), all of the aforementioned learning algorithms in 1) and 2) may ensure to reach one of the equilibria in the game. For cooperative games, which are usually featured by the process of bargaining or coalition formation among agents, the Nash bargaining solution can be learned through FP [37]. A team game is defined as the game in which the agents share the common payoff function, thus considered as a fully cooperative case of the general SG-based games. Since every team game can be modeled as a potential game [18], it is possible to apply best-response-based learning [77], stochastic FP [76] or no-regret learning [78] to learn the NE of a repeated team game. In the case of team SGs, each agent can also be associated with one single learning automaton at one game state. Then by applying L_{R-I} learning a pure-strategy NE is guaranteed to be reached [79].

D. A Summary of Model-Free Learning Algorithms

Before proceeding to the next section, we provide a summary of the learning mechanisms that have been introduced in this section in Figure 6. In Figure 6, the learning mechanisms
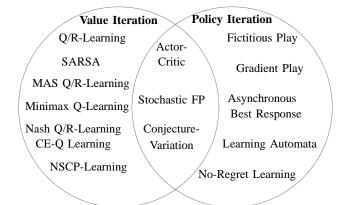
[6]About the definition of a Logit function, please refer to Section V-A3.

Fig. 6. A quick summary of the model-free learning algorithms.

TABLE VI
BRIEF CHARACTERISTICS OF MODEL-FREE LEARNING MECHANISMS

Learning Mechanism	System Model	Stability Property
Q/R-learning	Single-agent MDP	Optimality learning
SARSA	Single-agent MDP	Optimality learning
MAS Q/R-learning	Multi-agent MDP	Optimality learning
Minimax Q-learning	Noncooperative SGs	NE learning
Nash Q/R-learning	Noncooperative SGs	NE learning
CE Q-learning	Noncooperative SGs	CE learning
NSCP-learning	Noncooperative SGs	NE learning
FP	Noncooperative SGs/repeated games	ϵ-Equilibrium learning
Gradient play	Noncooperative repeated games	NE learning
Asynchronous Best Response	Noncooperative/Team repeated games	NE learning
LA	Noncooperative/Team repeated Games	ϵ-equilibrium learning
No-regret learning	Noncooperative/Team repeated games/SGs	NE/CE learning
Actor-critic learning	Single/Multi-agent MDP	Optimality learning
Stochastic FP	Noncooperative repeated games	NE learning
Conjecture-variation-based learning	Noncooperative SGs/repeated games	ϵ-equilibrium learning

are categorized according to the experience updating approach (i.e., value iteration or policy iteration) that they apply. In Table VI, we further summarize the characteristics of these learning mechanisms in terms of stability property and the system models (SAS, MAS and games) that they are built upon. Figure 6 and Table VI together provide a quick sketch of the algorithms that are to be surveyed with respect to their applications in cognitive wireless networks. More details of the characteristics of each learning mechanism will be provided in the following sections.

III. APPLICATIONS OF SINGLE-AGENT-BASED LEARNING IN COGNITIVE WIRELESS NETWORKS

Thanks to the property of self-organization, a model-free learner is able to reduce the level of required a-priori knowledge about the network model as well as the level of overhead due to explicit information exchange. It is also possible for the learner to adapt quickly to the changes of the network environment. As a result, model-free learning is particularly

TABLE VII
SUMMARY OF THE MAIN NOTATIONS IN SECTION II-A

Symbol	Meaning
$\mathcal{O}$	A finite set of observation states in a partially observable Markov decision process
o	A single observation state in a partially observable Markov decision process
$w(s)$	A weighting function to map a set of states s to a new state for state abstraction
$c_t(s,a)$ or c	Instantaneous cost function of a constrained MDP
λ	Lagrange multiplier

suitable for resource management and scheduling problems that demand self-exploration and self-organization of the network devices. Starting from this section, we will provide a comprehensive survey on the applications of model-free learning across different protocol layers in cognitive wireless networks following the broad-sense definition of CRNs. In a nutshell, the survey on the applications of model-free learning is organized based on the categorization of the learning mechanisms that is provided in Section II. According to the three types of mathematical models for decision-making, Sections III, IV and V are devoted to the applications of learning algorithms based on single-agent systems, loosely coupled multi-agent systems and game-based multi-agent systems, respectively. The notations used in this section are summarized in Table VII.

A. Applications of Learning in Single-Agent Systems

The early attempts in applying learning algorithms to wireless networking problems appeared even before the concept of cognitive radios was proposed. Generally, the a-priori knowledge of the environment evolution dynamics (e.g., the transition probabilities of the MDPs) is not required by the MDP-based, value-iteration learning schemes. Thus, the schemes are widely applied to the problems in the time-varying dynamics of the wireless environment that cannot be perfectly sensed. These problems include dynamic packet routing [80], Dynamic Channel Assignment (DCA) [81], [82] and joint radio resource management for multi-rate transmission control in WCDMA networks [83], just to mention a few. The strategy-learning schemes in these studies are featured by a single/centralized agent, and are usually based on the standard Q-learning algorithm given in (7). In early studies, the learning schemes are built upon the simplified system models. Thus, the issues such as the convergence conditions of the learning schemes are still not the focus of the discussion. As a result, the existence of Markovian property is simply assumed in most of these works [81]–[83]. Also, in order to reduce the complexity of the system model, the original MDPs modeling the network dynamics are usually transformed into new MDPs with reduced state-action space using state abstraction [84] or Q-table projection methods. However, the equivalence between the original MDPs and the re-transformed MDPs is generally not guaranteed (see the example of [83]). In most of these works (e.g., [80], [81]), the learning rules are designed in a heuristic manner. Sometimes the standard Q-learning schemes are modified by introducing the neural networks in order to represent the table of the state-action values and approximate the Q-value-updating function [80], [82], [83]. With these

simplifications, the convergence to an optimal strategy of the learning schemes in these studies is also not guaranteed.

Among different approaches for simplifying the MDP-based model of the network-control process, state abstraction [84] becomes a necessary way of trading off optimality for the efficiency of the single-agent-based learning mechanisms. The necessity of state-action-space reduction lies in the need for computational tractability of the learning schemes in the case of state-action-space explosion. This is especially necessary when a single agent is learning the strategy from a large set of candidate actions in a system with a huge number of states. In the context of networking problems, state abstraction maps an original network-control model based on one MDP into a new MDP with a smaller state-action set. Mathematically, state abstraction in MDPs can be defined as follows:

Definition 5 (State abstraction [84]). *For two MDPs $M = \langle \mathcal{S}, \mathcal{A}, u, \Pr(s'|s,a) \rangle$ and $\overline{M} = \langle \overline{\mathcal{S}}, \mathcal{A}, \overline{u}, \Pr(\overline{s}'|s,a) \rangle$, $\phi : \mathcal{S} \to \overline{\mathcal{S}}$ is such a mapping that $\{\phi^{-1}(\overline{s})|\overline{s} \in \overline{\mathcal{S}}\}$ partitions the state space $\mathcal{S}$. Define a weighting function $w : \mathcal{S} \to [0,1]$, where $\forall \overline{s} \in \overline{\mathcal{S}}, \sum_{s \in \phi^{-1}(\overline{s})} w(s) = 1$. $\overline{M}$ is an abstracted MDP of M, if the following conditions are satisfied:*

$$\overline{u}(\overline{s},a) = \sum_{s \in \phi^{-1}(\overline{s})} w(s)u(s,a), \tag{13}$$

and

$$\overline{\Pr}(\overline{s}'|\overline{s},a) = \sum_{s' \in \phi^{-1}(\overline{s}')} \sum_{s \in \phi^{-1}(\overline{s})} w(s)\Pr(s'|s,a). \tag{14}$$

However, the state-abstraction method generally requires that the state transition in the new MDP with reduced complexity to be well-defined. Namely, the linear-combination-based mapping in (13) and (14) needs to be established and the condition $\sum_{\overline{s}'} \overline{\Pr}(\overline{s}'|\overline{s},a) = 1$ needs to be satisfied. Since with model-free learning, the transition models are generally not known, it will be practically impossible to obtain an accurate model of the reduced MDP. In order to address such an issue, approximate abstraction is proposed in [85], [86]. In [85], [86], an on-policy reinforcement learning method, SARSA, is applied to the DCA problem in a multi-cell, multi-channel network with the consideration of handoffs. In the considered cellular network, N cells provide M channels to mobile stations, thus forming an $N \times (M+1) \times M$ state-action set. The arbitrary state-aggregation method proposed in [85], [86] aggregates the rarely encountered states by reducing the size of the channel state space to a fraction of the total number of the channels. The state variable representing the number of currently allocated channels is also excluded, which leads to a 98% reduction from the original state-action space. A more complicated state-action-space abstraction method can be found in [83]. It adopts the feature extraction method and maps the original state vector based on four dimensions, namely, the mean and variance of the interference from the existing connections, the transmission type and the required transmission rate, into a vector of the resultant interference profile. The feature extraction method is further adopted in stochastic-game-based modeling for strategy learning in CRNs [87], [88]. In [87], [88], the central spectrum moderator
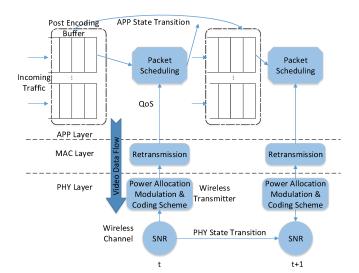
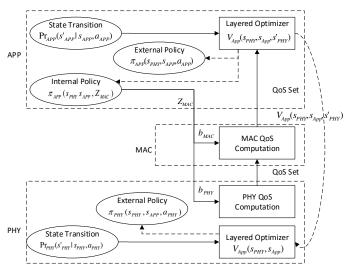Fig. 7. Real-time video streaming process (adapted from [89]).

Fig. 8. The operation and message exchange in the layered MDP (adapted from [89]).

allocates the transmission opportunities to the CRs through an iterative, second-price auction (see [18] for the definition of auctions), whose dynamics is jointly determined by the Signal-to-Noise-Ratio (SNR) of the channels and the buffer states of all the CRs. In [87], [88], multi-stage bidding is adopted. Since for each CR, the value of tax to be paid for using the channels are based on the inconvenience it causes to the other CRs, the individual CRs use their local tax announced by the central spectrum moderator to classify the channel-buffer states that the other (adversary) CRs are in. Therefore, individual CRs only need to exchange the pricing information with the central spectrum moderator, and no extra information exchange between the CRs is required. In these works, the feature extraction method does not only achieve the goal of state abstraction, but also help avoid the explicit information exchange between individual CRs.

With the development of MDP-based modeling in different protocol layers of the wireless networks (see examples in MAC layer [90], link layer [91] and application layer [92]), the SAS-based learning mechanisms in the cognitive wireless networks also gain more capabilities in addressing the radio

resource management problems. In [89], the problem of real-time video transmission over a single-hop, slow-varying flat fading channel is formulated as a systematic layered MDP (see Figure 7 and Figure 8 for a schematic view of the system and the corresponding layered MDP model). With the proposed problem formulation, the discrete system state is composed of three components, i.e., the SNR as the channel state in the PHY layer, the transmission opportunity as the state of the MAC layer and the amount of both the incoming traffic and the buffered packets as the state of the application layer (see Figure 8). The evolution of the joint state (s_{APP}, s_{PHY}) is modeled as a Markov chain controlled by the joint action $(a_{APP}, a_{MAC}, a_{PHY})$, in which a_{MAC} is composed of two internal actions b_{PHY} and b_{MAC}. The joint action is determined by the power allocation, the channel resource payment made to the spectrum moderator and the packet scheduling algorithm. The cross-layer management of packet transmission is formulated as a layered MDP. This is because for the Bellman optimality equation of the state value, the Dynamic Programming (DP) based expression can be decomposed into a two-loop DP-based optimization. In the two-loop optimization, it is assumed that both layers have access to the global state in each time slot. The inner loop (i.e., the application-layer optimization) only needs to know the joint MAC-application action and the reported state value of the PHY layer for policy updating, while the outer loop (i.e., the PHY-layer optimization) only needs to know PHY-layer action information and the reported state value from the application layer for policy updating. The layered Q-learning [93] can be applied to learn the optimal strategy for transmission, with the standard Q-value updating rule in (7) modified in each layer by incorporating the estimated Q-value from the other layer into the estimation of the local Q-values.

Apart from lacking the a-priori knowledge about the statistics of the underlying Markov process, the decision-making entity in the network may frequently face the constraints on the available resources. To tackle these constrained radio resource allocation/scheduling problems, the unconstrained MDP models are extended to the Constrained MDPs (CMDPs), based on which, modified reinforcement learning algorithms are also proposed [94]–[99]. Mathematically, a CMDP is defined by expanding the 4-tuple MDP model (Definition 1) to be a 5-tuple, $\langle \mathcal{S}, \mathcal{A}, u, c, \Pr(\mathbf{s}'|\mathbf{s}, a) \rangle$, with the additional cost/constraint element c [100]. Taking the average-reward CMDP as an example, a generic CMDP optimization problem can be stated as follows:

$$
\begin{aligned}
\max_{\pi} \quad & h^{\pi}(s) = \lim_{T \to \infty} \sup \frac{1}{T} E_{\pi} \left\{ \sum_{t=0}^{T-1} u_t(s_t, a_t) \right\}, \\
\text{s.t.} \quad & C^{\pi}(s) = \lim_{T \to \infty} \sup \frac{1}{T} E_{\pi} \left\{ \sum_{t=0}^{T-1} c_t(s_t, a_t) \right\} \leq C_{\max}.
\end{aligned}
$$
(15)

According to Theorem 12.7 of [100], we have the following theorem for the average-reward CMDP:

Theorem 1. *If the underlying Markov chain of the CMDP, $\langle \mathcal{S}, \mathcal{A}, u, c, \Pr(\mathbf{s}'|\mathbf{s}, a) \rangle$, is unichain and the sequence of the immediate cost c_t is bounded below and satisfies the following*

growth condition:

> *for* $c : \mathcal{K} \to \mathbb{R}$ *there exists a sequence of increasing compact subsets* $\mathcal{K}_i$ *of* $\mathcal{K}$ *such that* $\cup_i \mathcal{K}_i = \mathcal{K}$ *and* $\lim_{i \to \infty} \inf\{c(\kappa); \kappa \notin \mathcal{K}_i\} = \infty$,

then there exists an optimal Lagrange multiplier λ^* *such that the optimal solution of the CMDP is equivalent to the optimal solution of the unconstrained MDP,* $\langle \mathcal{S}, \mathcal{A}, g = u - \lambda^* c, \Pr(\mathbf{s}'|\mathbf{s}, a) \rangle$.

According to Theorem 1, the non-structured learning schemes for the unconstrained MDP based on the Lagrangian dual function can be developed for solving the resource management/scheduling problems in the form of both R-learning [95], [97], [99] and Q-learning [94], [98], [101], depending on the form of the reward/cost of the CMDP. Apart from the primal-dual equivalence based solution, it is also possible to develop constrained learning algorithms by exploiting the structure of the specific problems. The special structure is featured by the convexity of the objective and constraint functions in the original CMDP, or the modularity of the objective or the constraint functions [98], [102]. When certain structural property of the network control problems is satisfied (specifically, when both the instantaneous payoff and the constraint cost are multi-modular), the constrained structured-learning algorithm can be applied in the form of primal projection or submodular parameterization [102].

In addition to not knowing the environment evolution dynamics and being limited by the resource constraints, the learning agents in a wireless network may also lack the ability of complete state-information acquisition. This can be a common issue in scenarios such as DSA networks, in which the secondary devices lack the capability of performing full-spectrum sensing due to the limited number of antennas [109]. The common approach to handle such a problem is to model the radio resource management problem as a Partially Observable Markov Decision Process (POMDP). Extending from Definition 1, an unconstrained POMDP can be defined as a 6-tuple, $\langle \mathcal{S}, \mathcal{O}, \mathcal{A}, u, \Pr(s'|s, a), \Pr(o|s, a) \rangle$, in which $\mathcal{O}$ is the set of observations o, and $\Pr(o|s, a)$ denotes the mapping probability between the system states and the observations. Instead of directly observing the state information of s, the learning agent can only obtain the network observation o. In the POMDPs, the random process associated with the observation is no longer a Markov process. A standard model-based solution to the POMDP is to convert the recorded state observations into belief states, and obtain a new unconstrained MDP with a continuous state space of the belief states. However, when the state-transition and the state-observation mapping is unknown, the TD-based learning schemes cannot be directly used for learning the optimal strategies of the POMDPs. Instead, other learning algorithms such as actor-critic learning [110] and policy-gradient-based learning [111] are applied. In [105], a delay-constrained least-cost routing problem in MANETs is modeled as a POMDP, the belief state of which captures the link-delay uncertainty due to the imprecise link state information. The belief-policy mapping is considered as a parametric function, the policy parameter of which is learned through a standard actor-critic learning

method. In [107], to solve the DSA problem in a CRN, the channel access process of the Secondary Users (SUs) is first modeled as a constrained POMDP. In the constrained POMDP, a reward function is used to collect the instantaneous reward of the SUs, while a cost function reflects the instantaneous cost of the Primary Users (PUs) due to the channel interference from the SUs. The partial observation in the problem comes from the imperfect spectrum sensing of the SUs over the primary channel state. After converting the original constrained POMDP into an unconstrained POMDP with the help of the Lagrange multiplier, the learning algorithm based on policy gradient [111] is applied for finding a local optimal policy.

To summarize this section, we categorize in Table VIII the aforementioned works (and some more) on SAS learning according to the networking applications that they focus on. As shown by Table VIII, the SAS-based learning algorithms are powerful in addressing a number of radio resource allocation problems, as long as they can be formulated as a single-link-centric one. However, it is worth noting that although the theoretical support for the convergence of the SAS-learning schemes has been well studied, such an issue still needs to be addressed under practical circumstances.

IV. APPLICATIONS OF LEARNING BASED ON LOOSELY COUPLED MULTI-AGENT SYSTEMS

The multi-agent learning scheme naturally leads to the framework of distributed decision making, thus the possibility of self-organization without a dedicated central coordinator. Therefore, it is considered especially appropriate for the network management problems in the CRNs, device-to-device (D2D) networks, heterogeneous networks (HETNETs) and ad-hoc networks, as long as the networks consist of multiple independent decision-making entities. However, although the framework of distributed decision making naturally leads to the consideration of adopting the multi-agent decision learning scheme for network control, it is worth noting that for most cases it may be difficult to directly adopt the learning mechanisms based on the loosely coupled MAS by simply ignoring the interactions between the network entities and treat each of them as an independent learner. Due to the existence of device interaction, it is necessary to carefully investigate into both the advantage and the limitation of formulating a distributed network control problem as a loosely coupled MAS. Furthermore, when adopting the model of learning in the loosely coupled MAS, it is still necessary to check to what level the information exchange between the learning agents is needed, and in what ways it can help improving the performance of the network.

The new notations used in this section are summarized by Table IX.

A. Applications of Distributed Learning Based on the Model of Loosely Coupled Multi-Agent Systems

For distributed learning in wireless networks, it is usually difficult to definitely classify between a non-game-based, multi-agent decision learning scheme and an SG-based learning scheme. The reason for this lies in the inherited nature of

TABLE VIII
APPLICATIONS OF SAS-BASED LEARNING SCHEMES IN COGNITIVE WIRELESS NETWORKS: A SUMMARY

Network Type	Application	Problem Formulation	Reference	Learning Scheme	Learning Scheme Variation	Convergence
Cellular	Dynamic channel allocation	MDP [81], [82], [85], [86], semi-MDP [103]	[81], [82], [85], [86], [103]	Q-learning [81], [82], [103], SARSA [85], [86]	Neural Network [82], State abstraction [85], [86], N/A [103]	N/A
	Multirate transmission control	MDP	[83]	Q-learning	Neural network with feature extraction [83]	N/A
	Call admission control	CMDP	[94], [101]	Q-learning [94], [101]	State abstraction [94], [101]	N/A
	Joint admission-bandwidth control	CMDP	[99], [104]	Q-learning [104], Neural network [99]	N/A	N/A
Single link	Cross-layer resource allocation	Layered-MDP	[89], [93]	Layered Q-learning [89]	Virtual experience tuples [93]	N/A
	Scheduling-admission control	CMDP	[96]	Stochastic sub-gradient	N/A	Deterministic optimal policy [96]
	V-BLAST power-rate control	CMDP	[102]	Q-learning	Constrained structured Q-learning	Randomized optimal policy
MANETs	QoS routing	POMDP	[105]	Actor-critic learning	N/A	N/A
CRNs	Dynamic spectrum access	MDP [106], CMDP [95], [97], POMDP [107]	[95], [97], [106], [107]	Actor-critic learning [106], R-learning [95], [97], policy gradient [107]	N/A [95], [106], [107], Arbitrary state reduction [97]	Deterministic optimal policy [95], N/A [97], [106], Local optimum policy [107]
HETNETs	Vertical handoff	CMDP	[98]	Q-learning	N/A	Optimal randomized policy
	Admission control	MDP	[108]	Q-learning	Q-learning based on neural-fuzzy-inference network	N/A

TABLE IX
SUMMARY OF THE NEW NOTATIONS IN SECTION IV

Symbol	Meaning
$\gamma_r^{i,F}$	The received SINR for femto/pico link i over resource block r
P_r^{iF}	The transmit power of femto/pico BS
$g_{ii,r}^{FF}$	The link gain between the femto/pico BS and its user
$g_{ii,r}^{MF}$	The link gain between the macro BS and the femto/pico user
σ^2	noise power
$I[x]$ or $I(x,y)$	The indicator function
$w_i(j)$ or $w_i'(j)$	The weight assigned by agent i for its neighbor j's instantaneous reward or estimated state value
Y	The social reward of a group of agents
y	The private reward that an individual agent chooses

strategy coupling in most of the practical networking problem setups. One typical example is illustrated in [113], [114], which consider that L macrocells and N femtocells/picocells operate over the same frequency band (see Figure 9) in a HETNET. In order to develop a self-organized power allocation scheme for the downlink transmission in the HETNET, the Shannon capacity of a link is considered as the individual utility of a cell, which is a function of the Signal-to-Interference-plus-Noise-Ratio (SINR) of the transmitting link in that cell. Take the femtocells/picocells as an example, when both the intra-cell interference and the cross-tier interference are considered, for femotocell/picocell link i, the SINR at the receiver is determined as follows:

$$\gamma_r^{i,F} = \frac{P_r^{i,F} g_{ii,r}^{FF}}{\sum_{j=1}^{L} P_r^{j,M} g_{ji,r}^{MF} + \sum_{k=1,k\neq i}^{N} P_r^{i,F} g_{ki,r}^{FF} + \sigma^2}, \quad (16)$$

where $P_r^{i,F}$ is the transmit power of femto/pico Base Station (BS) i over the resource block r, $g_{ii,r}^{FF}$ is the link gain between
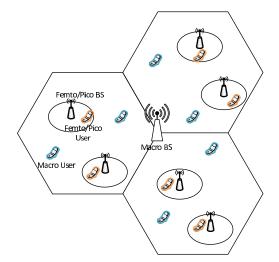

Fig. 9. Structure of a HETNET with both inter-cell and cross-layer interference. A HETNET is featured by the hierarchy in the network structure, which is comprised by the high-power, high-capacity, wide-range macrocells and the low-power, low-capacity, small-range femtocells/picocells [112].

the femto/pico BS and its user, $g_{ji,r}^{MF}$ is the link gain between macro BS j and the femto/pico user, $g_{ki,r}^{FF}$ is the link gain from another femto/pico BS k to the user of femto/pico BS i, and σ^2 is the noise power.

Apparently, the capacity of femto/pico link i is determined not only by the transmit power of femto/pico BS i, $P_r^{i,F}$, but also by the inter-cell interference $\sum_{k=1,k\neq i}^{N} P_r^{i,F} g_{ki,r}^{FF}$ and the cross-tier interference $\sum_{j=1}^{L} P_r^{j,M} g_{ji,r}^{MF}$. Therefore, the private utility of femto/pico link i is also a function of the strategy of the other femto/pico BS k ($k = 1, \ldots, n, k \neq i$) and all the macro BSs j ($j = 1, \ldots, L$). The goal of the local cells for maximizing the individual utilities conflicts with each other,

and it is difficult to decompose the strategy coupling between the cells. As a result, many works formulate the same problem as a noncooperative repeated game [115], [116]. However, it is still possible to tackle such a power control problem by treating the strategies of the other BSs as part of the environment dynamics. For example, in [113] the system state from the perspective of a femto/pico link is designed as a binary one:

$$I_t^{i,r} = \begin{cases} 1, & \text{if } \gamma_{r,t}^{i,F} < \gamma_{\text{Th}}, \\ 0, & \text{otherwise,} \end{cases} \quad (17)$$

which is based on hard thresholding (compared with the permitted SINR given as γ_{Th}) of the macrocell user with the interference from the femto/pico links. The similar network-state formulation can be found in the other works such as [114]. By adopting a standard Q-learning scheme based on the assumption of independent state-value evolution, it is assumed in [113], [114] that the dynamics of the aggregated interference to the macrocell user is a stationary Markov process. Consequently all the strategies of the other femto/pico users are treated as stationary ones hence part of the wireless environment. In most of the cases, such a formulation/solution with the distributed MDPs and independent Q-learning algorithm may not guarantee the convergence to any equilibrium. However, empirical studies show that when using the distributed Q-learning scheme, convergence can still be achieved given a sufficiently large number of iterations [113], [117], [118], and the distributed Q-learning algorithm is also able to achieve a better performance compared with the non-adaptive algorithms [114], [117], [118]. Although not mathematically proved, one possible explanation for such a result may lie in Proposition 2, since one independent Q-learning agent is always able to converge as long as the other agents happen to converge in behavior.

Generally, for the network management problems with strategy coupling, directly adopting the distributed learning schemes in the loosely coupled MAS (e.g., multi-agent learning in the form of distributed, independent Q-learning) can be considered as an approach that trades off the certainty of algorithm convergence for the simplicity of system analysis and learning-rule design. Except for heterogeneous networks, applications that follow such a design pattern can be found in the problem formulation such as distributed DSA with the SU collisions [119]–[121], power allocation in the overlay, cognitive wireless mesh network [122] and dynamic spectrum management in 4G cellular networks [123]. Although with many studies that adopt such a design pattern for the learning schemes, it is important to reiterate that overlooking strategy coupling may result in poor performance of each learning agent. In [124], a problem of DSA management with 2 SUs over 2 primary channels (see Figure 10) is used to exemplify how the lack of coordination between individual agents may impact the agent performance. In [124], the availability of a primary channel is modeled as a two-state discrete Markov chain. The SUs try to access the idle primary channel while avoiding the collision with the other SU. The adaptation of the channel-access strategies is formulated as a POMDP, in which the observation of an SU includes 3 states: busy, collision and
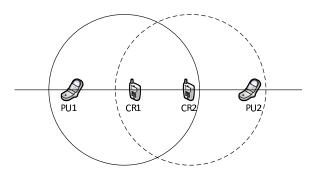

Fig. 10. Illustration of the interference map for the two-SU-two-PU DSA network [124].

success. Based on the assumption that the presence of the other SU can be ignored, a model-based single-user approach for strategy updating is proposed. When compared with the cooperative approach, which allows the SUs to exchange their belief state vectors of the POMDP, the performance of the single-user-based approach is shown to be significantly inferior. Moreover, the simulation results in [124] show that the performance of the single-agent-based approach is even worse than that of the deterministic channel-assignment scheme, which indicates that in the situation of strategy coupling, allowing some degree of cooperation will be essential.

In order to balance between the simplicity of the learning mechanism (namely, the distributiveness of strategy learning) and the optimality of the learning algorithm, careful modeling is needed with respect to different network scenarios. In [125], a set of decision-learning mechanisms based on distributed Q-learning is adopted for a scalable DSA mechanism in an overlay CRN. The goal of the learning mechanism design is to obtain the near-optimal strategies without the explicit coordination among the SUs. It is shown in [125] that by properly designing the private/local objective functions of the individual SUs, the needs of both agent coordination and distributed decision-learning can be fulfilled. In [125], the SUs are assumed to share the temporarily free band roughly equally. It means that the reward of an individual SU with DSA, $u_i(t)$, is approximately equal to the average of the social reward of all the SUs that attempt to use the same primary band (denoted by $Y(t)$):

$$u_i(t) = \frac{1}{|\mathcal{N}_i(t)|+1} Y(t) = \frac{1}{|\mathcal{N}_i(t)|+1} \sum_{i=1}^{|\mathcal{N}_i(t)|+1} u_i(\mathcal{N}_i(t)), \quad (18)$$

where $\mathcal{N}_i(t)$ is the set of SUs that interfere with SU i over the same band at time t. The PU activity is also modeled as a two-state Markov chain. In [125], two guidelines are proposed for designing the private/individual objective function of each SU:

1) *alignedness*, which reflects agent coordination, and the full alignedness requires the SUs not working against each other when maximizing their own private objectives;

2) *sensitivity*, which reflects the efficiency of the individual learning processes and requires the SUs to be able to discern the impact of their own action changes so as to learn about the better local strategies fast enough.

In [125], the measurable indices of "factoredness" and "learnability" are introduced to measure alignedness and sen-

sitivity of the private objective function, respectively. Denoting the selected private objective function as y_i (y_i may not be the same as u_i) and the joint deterministic strategy by the SUs over the same band as $\boldsymbol{\pi} = (\pi_i, \pi_{-i})$, the degree of factoredness and learnability can be expressed as in (19) and (20), respectively:

$$F_{y_i} = \frac{\sum_{\pi_i} \sum_{\pi_{-i}} I[(y_i(\boldsymbol{\pi}) - y_i(\boldsymbol{\pi}'))\,(Y(\boldsymbol{\pi}) - Y(\boldsymbol{\pi}'))]}{\sum_{\pi_i} \sum_{\pi_{-i}} 1}, \quad (19)$$

$$L_{i,y_i}(\boldsymbol{\pi}) = \frac{E_{\pi'}[|y_i(\boldsymbol{\pi}) - y_i(\pi_{-i}, \pi_i')|]}{E_{\pi'_{-i}}[|y_i(\boldsymbol{\pi}) - y_i(\pi_{-i}, \pi_i')|]}, \quad (20)$$

where $I[x]$ is the indicator function, $I[x] = 1$ if $x > 0$ and $I[x] = 0$ otherwise, and F_{y_i} ($0 \le F_{y_i} \le 1$) measures the consistence between the local objective and the social payoff. The higher the degree of factoredness (i.e., the value of F_{y_i} is, the more likely a change of the local action by SU i will have the same impact on both its private reward and the global reward. L_{i,y_i} measures the sensitivity of local reward to the local action changes. According to (20), the higher the sensitivity (i.e., the learnability), the more the dependence of $y_i(\boldsymbol{\pi})$ on the local actions of SU i. By employing the property described by (18), namely, the private reward $y_i(t) = u_i(t)$ being proportional to the global reward $Y(t)$, it is shown in [125] that a good objective function can be obtained by removing from $Y(t)$ the effects of all SUs other than SU i. A general form of such a local objective function can be expressed as follows:

$$D_i(\boldsymbol{\pi}) = Y(\boldsymbol{\pi}) - Y(\pi_{-i}). \quad (21)$$

Since $u_i(t)$ is a function of both $Y(t)$ and the cardinality of the interfering-SU set $\mathcal{N}_i(t)$, all that SU i needs to obtain the value of D_i is to estimate $|\mathcal{N}_i(t)|$ given the information that SU i observes locally. It is shown that with the proposed objective function (21), the distributed learning scheme achieves better spectrum efficiency than those learning with both private reward and global reward. From the game theoretic perspective, spectrum access with the individual reward as in (18) can be interpreted as a cardinal potential game [18], in which (21) is in the exact form of a potential function. In this sense, the design of the objective function in [125] can be considered as a special case of global-reward-based learning, and may not be easily extended to a general radio resource management problem such as [113], [114]. Although the two indices in (19) and (20) provide important guidelines on individual utility function design for distributed learning, it is still needed to find appropriate approaches other than that given by (18) for the networking applications which cannot be modeled as a potential game.

Instead of designing a different objective function, the learning scheme itself can also be tailored to meet the requirement of radio resource management. One example of learning scheme design in the strategy-coupling scenario is provided by [126], which studies an Aloha-like spectrum access scheme without any negotiation in a multi-user, multi-channel CRN (Figure 11). In [126], N primary channels are modeled as N independent, two-state Markov chains, while the SUs are assumed to have no mutual communication and need to learn the collision-avoidance strategies online. Instead of adopting
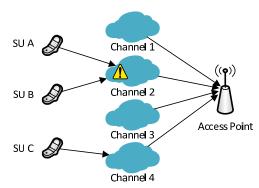

Fig. 11. Channel access competition and conflict in an Aloha-like multi-user-multi-channel CR system [126].

the standard state-value evolution model given in (4) and the TD-based strategy-learning mechanism given in (7), the expected one-time reward is adopted as (22):

$$Q_{ij}^{\mathbf{s}} = E[u_i | a_i(t) = j, \mathbf{s}(t) = \mathbf{s}], \quad (22)$$

and a learning mechanism without considering the future reward is designed as (23):

$$Q_{ij}^{\mathbf{s}}(t+1) = (1 - \alpha_{ij}(t))Q_{ij}^{\mathbf{s}}(t) + \alpha_{ij}(t)u_i(t)I(a_i(t), j). \quad (23)$$

In (22) and (23), $a_i(t) = j$ represents the action of SU i to select channel j for transmission, $\mathbf{s}$ is the vector of the channel states, $\alpha_{ij}(t)$ is the learning step, $u_i(t)$ is the instantaneous reward of SU i and $I(x, y)$ is the indicator function (i.e., $I(x, y) = 0$ if $x \ne y$ and $I(x, y) = 1$ if $x = y$). Although (23) appears in a similar form to distributed Q-learning, it is derived based on the analysis of the channel contention as an SG. It is shown in [126] that with the Boltzmann distribution-based strategy exploration, the learning scheme in (23) is equivalent to the Robbins-Monro iteration [127] and converges asymptotically to a stationary point (i.e., an NE) with probability one.

Generally, the aforementioned multi-agent learning schemes can be divided into two categories, namely, distributed learning based on the assumption of purely independent state-value evolution (e.g., [113], [114], [117]–[123]) and distributed learning based on the structural property of the specific resource management problems (e.g., [125], [126]). Although both of them do not require explicit information exchange among network devices, sometimes introducing a certain level of information exchange (at the cost of more overhead) can help improve the network performance. In the literature, the learning schemes with explicit information exchange is usually referred to as learning based on Distributed Value Function (DVF). With DVF, local devices are required to share their state-value/reward functions with the neighbors. Instead of learning the Q-value based on the individual reward or local state values, individual decision making aims at the maximization of both the local and the neighbors' weighted sum of rewards/state-values. By modifying (7), a typical learning mechanism with DVF can be expressed as

$$Q_i^{t+1}(s_i, a_i) \leftarrow (1 - \alpha_t)Q_i^t(s_i, a_i) + \alpha_t \left(u_i^t(s_i, a_i) + \beta \sum_{j \in \mathcal{N}(i)} w_i(j)V_j(s_j) \right), \quad (24)$$

TABLE X
APPLICATIONS OF INDEPENDENT-LEARNER LEARNING SCHEMES IN COGNITIVE WIRELESS NETWORKS: A SUMMARY

Network Type	Application	Reference	Strategy Coupling Assumptions	Learning Scheme	Convergence
CRNs	Aggregated interference control	[117]	None	Independent Q-learning for MDP and neural network for POMDP [117]	N/A
	Joint spectrum and power management	[119], [122]	None [119], coupling as a noncooperative game [122]	Independent Q-learning [119], [122]	N/A
	Dynamic spectrum access	[120], [123], [125], [126]	None [120], [123], coupling with fully connected topology [125], Coupling as a noncooperative game [126]	Independent Q-learning [120], Win-or-Learn-Fast (WoLF) [123], Independent learning (unspecified) [125], Modified independent Q-learning without considering the future states [126]	N/A [120], [123], near optimal strategy [125] or NE [126]
	Joint sensing-time and power allocation	[121]	None	Independent Q-learning	N/A
HETNETs	Femto-user power allocation	[113]	None	Independent Q-learning	N/A
	Inter-cell interference coordination	[114]	None	Independent Q-learning	N/A
Sensor networks	Coverage and energy consumption management	[128]	Coordinated decision-making	DVF-based learning	N/A
Cooperative networks	Power and relaying probability management	[129]	Coordinated decision-making	Q-learning based on distributed reward and value function,	Local optimal point
Cellular networks	Power allocation and experience sharing	[130]	Coordinated decision-making	DVF-based learning	N/A

in which $\mathcal{N}(i)$ is the set of device i's neighbors (including i) and $w_i(j)$ is the weight that determines the contribution of device j's state-value to device i's estimation of V_i.

The applications of the DVF-based learning mechanism in wireless networks can be found in [128]–[130]. In [128], DVF-based learning is used in an ad-hoc sensor network to coordinate the sensing and hibernation operation as the state of the grid-point coverage changes. To encourage the sensor node with a larger coverage area to perform the sensing operation, the individual reward is designed as a function of the number of the covered grid points. It is shown that DVF-based learning outperforms the independent learner-based learning algorithm, especially under the condition of high sensor node densities. In [129], a learning algorithm based on the exchange of both the instantaneous reward and the estimated local state-value is proposed for the joint power control and relay selection in a distributed cooperative network. The proposed learning scheme is featured by weighting over both the instantaneous reward and the estimated local state-value that are shared by the neighbor nodes, and thus is called learning with the Distributed Reward and Value (DRV) function. By extending (24), the rule of learning with DRV can be expressed as follows:

$$Q_i^{t+1}(s_i, a_i) \leftarrow (1-\alpha_t)Q_i^t(s_i, a_i) +$$
$$\alpha_t \left(\sum_{j \in \mathcal{N}(i)} w_i'(j) u_j^t(s_j, a_j) + \beta \sum_{j \in \mathcal{N}(i)} w_i(j) V_j(s_j) \right), \quad (25)$$

in which $w_i'(j)$ and $w_i(j)$ are the weight of node i given to its neighbor j's instantaneous reward and estimated state value, respectively. With the learning scheme given in (25), each node in the network maintains a vector of both the channel/buffer state of its direct link and the channel/buffer state of its cooperative link. It is shown in [129] that learning

based on sharing both the instantaneous rewards and the local state values can achieve a better power efficiency than that using only the local reward or the local state value information. In [130], the DVF-based learning scheme is adopted in a real-time multimedia cellular network to adapt the power allocation of interfering links. In addition to coordinating the individual links, the Q-value updating mechanism (24) is also used to improve the convergence of the newly adopted links in the network.

In Table X, we categorize the works discussed in this subsection according to their respective applications. For applications of multi-agent independent-learning schemes in wireless networks, convergence of learning remains an open issue in most of the existing studies. Compared with the SAS-based learning algorithms, adopting independent learning schemes requires more attention for any specific networking optimization problem.

B. Experience Sharing Based on Distributed Learning

Apart from improving the expected network performance with shared information in the form of structured reward/state-value functions (e.g., using the social reward and the DVF/DRV functions), another consideration in MAS-based learning is whether information sharing can also help the individual learning agents to speed up their learning processes. To answer this question, it is necessary to investigate into the homogeneity of the distributed learning processes so that we can check whether one learning process may be able to benefit from the "shared experience" offered by another learning process, and furthermore, in what form such a "shared experience" would be.

We call a group of distributed learning processes homogeneous when the distributed learning agents apply the same
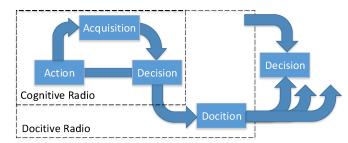
Fig. 12. Docitive cycle which extends the cognitive cycle by cooperative teaching (adapted from [133]).

learning method with an evolution determined by exactly the same stochastic process. In the framework of homogeneous learning processes, it is possible for individual agents to share their private experience (e.g., strategies, estimated Q-values) with the other agent in order to accelerate the learning process and improve the performance. Recently, the possibility of applying the teacher-pupil paradigm in human cognition to solve the wireless networking problems has been discussed in a series of studies [131]–[134]. In these pioneering studies, the paradigm of "docitive network" was proposed based on the extension of distributed cognitive networks (Figure 12). In the framework of docitive networks, "docition" (teaching) is performed by a more experienced network agent to accelerate the learning process of the other agents. Depending on the degree of docition among the wireless devices, the teaching-learning process can be distinguished into 3 categories [131]:

- *Startup docition*: each wireless node learns independently. When a new node joins the network, instead of learning from zero experience, it learns the policies from docitive nodes which have already acquired a certain level of expertise on strategy selection.
- *Adaptive docition*: the nodes exchange information about the performance of their learning processes. The docitive nodes share policies and the learning nodes learn from the expert neighbors which have the best performance.
- *Perfect docition*: each node in the network is able to observe the joint action and all individual rewards. Based on the observation, every docitive node models its interaction with the rest of the network as a complete centralized MDP separately and selects its individual actions.

The basic prerequisite for implementing docition in any networking problems is that the individual learning processes can be modeled as parallel, homogeneous MDPs, through which imitating the strategies of the docitive nodes by the learning nodes will not influence the policies of the docitive nodes. However, empirical studies have shown that relaxing such a constraint in the situation of a noncooperative game-like scenario may also help improve the performance of the learning nodes [132], [134], [135]. In [132], the distributed downlink power allocation problem in an IEEE 802.22 WRAN (underlay to the TV-Broadcasting bandwidth) is studied. An aggregated interference model from the SUs to the PU is considered. The channel state experienced by the individual SUs is defined by a binary state according to hard thresholding on the aggregated interference, which is similar to (17). Each secondary BS ignores the impact of the other BSs on the channel state and adopts a standard independent Q-learning scheme to learn its

own power selection strategy. The docition process is based on exchanging the Q-tables among the neighbor secondary BSs. In this case, the learning nodes perform either the startup docition or the adaptive docition periodically by adopting the Q-tables of the expert nodes with the best performance. The simulations in [132] show that the docitive paradigm significantly speeds up the learning process with respect to the case of independent learners. A similar approach is adopted in [134], [135], which study the power allocation problem in self-organized heterogeneous networks with femotocells. In these studies, a cross-tier interference model is adopted in a manner similar to (16), while strategy coupling among the femto links is also ignored by individual learners. Again, here docition is performed through exchanging the Q-tables among the neighbor nodes. In [134], the similarity metric to measure of the correlation between the femto BS strategy and the aggregated interference to the macrocell is introduced as a user-defined gradient. The proposed metric measures the similarity of the policies between two neighbor nodes. With the similarity metric, the learning nodes can not only adopt the Q-tables from the neighbor nodes with the best performance, but also take into account the degree of the similarity between their own action-state correlation and their neighbors'.

While it is relatively easy to implement docition in the framework of independent Q-learning based on the model of parallel, homogeneous MDPs, it generally remains an open issue to estimate the similarity of the policies between two neighbor learners when the learning processes are heterogeneous. Especially, in the scenario of strategy coupling and interest conflict, imitating the strategies or the Q-tables of the adversary neighbor node with the best performance may result in strategy oscillation. Such a situation can be illustrated by revisiting the power allocation problem defined by (16). In the simplified situation of mutual interference with only two femto BSs, increasing the transmit power of one BS will result in the performance deterioration for the other BS, because the interference to the other BS is also increased. Consider the case that the BS with the smaller transmit power decides to adopt the strategy of its rival BS by increasing its transmit power. If independent Q-learning is used by both BSs to learn their power selection strategies, the other BS will soon discover that it will benefit from increasing its current transmit power too. This creates an "arm race" situation in which each BS begins to increase its transmit power in turn until both the BSs reach their maximum power level, which is a typical situation of the prisoner's dilemma in noncooperative games. Such an unwanted situation can be avoided if both BSs treat the power allocation process as a noncooperative game and adopt the learning methods in games such as Fictitious Play (FP) and best response without any docition procedure[7]. As a result, in works such as [137] the docitive paradigm and the game-based learning paradigm are considered two controversial frameworks for strategy learning. However, it is worth noting that with emerging techniques such as transfer

---

[7]Studies adopting the same mutual interference model as in (16) within the framework of repeated games can be found in [136]. In [136], the best response without docition ensures the convergence to the Pareto dominant equilibria.

TABLE XI
SUMMARY OF THE MAIN NOTATIONS IN SECTION V

Symbol	Meaning
$\kappa_i^t(a_{-i})$	The statistic of player i for its opponent's actions
$\theta_i^t(a_{-i})$	The estimated probability of player i for its opponent to play action a_{-i}
BR$(\cdot)$	The set of best-response actions
ν	The learning parameter for perturbation in SFP
α_t	The learning factor in SFP
$q_i^t(a)$	The estimated frequency of local actions
ϵ_t	The time-varying step size in GP
μ, μ_t	The learning parameters in GP, LA and no-external-regret learning
p_i	The probability of accessing a channel in a random medium access game
R_m	The transmit rate over channel m
$R_i(\pi_i, a_i\|a_{-i})$	The regret of player i for playing strategy π instead of playing a_i
$R_i^t(a_i^t, a_i')$	The regret for agent i not playing a_i' at time t
$c_i^t(\mathbf{s}, a_{-i})$	The conjecture of opponent policy $\pi_{-i}(\mathbf{s})$ by agent i at time t
$\overline{\pi}_i^t(\mathbf{s}, a_i)$	The reference point for conjecture learning

learning [138] and experience-weighted attraction learning [139], incorporating the teaching process in the game-based framework of learning is no longer impossible. For this part, we will leave the discussion of more details to Section VI.

V. APPLICATIONS OF GAME-BASED LEARNING IN COGNITIVE WIRELESS NETWORKS

Generally, there are the limitations of the distributed learning mechanisms (e.g., the algorithms reviewed in IV) that post the necessity of introducing the game-based learning mechanisms in CRNs. By modeling distributed network control problems as games, it is possible to better address the problems raised by device interactions in the networks. Also, it is possible to design learning schemes that theoretically guarantee the convergence of the individual strategies to a fixed point or equilibrium, while such convergence is usually not guaranteed by the distributed learning mechanisms. In this section, we consider the repeated games as the special cases of SGs and introduce the applications of learning algorithms based on repeated games and SGs separately. We will organize the learning algorithms based on the three game property dimensions discussed in Section II-C. Our major focus will be (a) the rules in each learning scheme; (b) the conditions and properties of the games with which a specific learning scheme may converge; and (c) the degree of information exchange required by each learning scheme to achieve convergence. The new notations used by this section can be found in Table XI.

A. Applications of Learning in the Context of Repeated Games

Repeated games play an important role in problem formulation for distributed network control. When the network evolution is not subject to a stochastic environment, most of the network control problems that requires considering the interactions among distributed devices can be formulated as a repeated game instead of an MAMDP. In contrast to the MDP-based learning mechanisms that heavily depend on value iteration, policy iteration now plays an important role in deriving the learning rules for repeated games. In the context of repeated games, model-free learning emphasizes

more on the situation of information locality. This is because in many practical scenarios, the information of the local utilities, actions or strategies of one network device may not be available to the other devices due to either the concern of privacy or the lack of enough resources for information exchange. In this subsection, we will organize our survey on the applications of learning in repeated games according to the prototypical learning schemes that they are based on. These prototypical learning schemes include (i) fictitious play, (ii) gradient play, (iii) learning automata and (iv) no-regret learning.

1) Fictitious Play and Stochastic Fictitious Play: The basic prerequisite of the standard FP is that the agents are willing to reveal their (discrete) action information to the others after each round of play, so they can track the frequency of action selection by the other agents [33]. With FP, agent i assesses the distribution of its opponent's actions at round t as follows:

$$\kappa_i^t(a_{-i}) = \kappa_i^{t-1}(a_{-i}) + I(a_{-i}^{t-1}, a_{-i}). \tag{26}$$

Agent i estimates the probability for the opponent agents to play the joint action a_{-i} at round t as:

$$\theta_i^t(a_{-i}) = \frac{\kappa_i^t(a_{-i})}{\sum_{a_{-i}' \in \mathcal{A}_{-i}} \kappa_i^t(a_{-i}')}. \tag{27}$$

In this sense, FP is sometimes considered as a model-based learning mechanism since with (27) it tries to build the model of the opponents' joint policy from accumulated experience. However, compared with other model-based, non-learning mechanisms such as dynamic programming for MDPs, FP does not need any a-priori knowledge of the system or other players. Based on (27), FP is defined as any rule that assigns the best response to agent i given its current estimation of the opponent policy $\theta_i^t(a_{-i})$. Usually, such an operation is represented by $a_i^t(a_{-i}) \in \mathrm{BR}_i(\theta_i^t(a_{-i}))$, where the operator BR$(\cdot)$ derives the best-response action set. Typically, BR$_i$ can be derived by maximizing the estimated expected payoff of agent i: $\mathrm{BR}_i(\theta_i^t(a_{-i})) = \arg\max_{a \in \mathcal{A}_i} E[u_i(a, \theta_i^t(a_{-i}))]$. The convergence property of FP in a general repeated game is given by Theorem 2 [33].

Theorem 2 (Convergence of FP). *1) Strict NE[8] are the absorbing state for the process of fictitious play. 2) Any pure-strategy steady state of fictitious play must be an NE.*

Theorem 2 gives the sufficient condition for FP to converge to an NE. Thereby, the convergence of FP-based learning is guaranteed in any repeated games that possess at least one pure-strategy NE. According to Theorem 2, a typical way of checking the convergence condition for FP in a game is to check if the game possesses certain properties (such as being potential or S-modular [18]) that guarantee the existence of a pure-strategy NE.

As long as the learning agents are able to observe the actions of the rival agents or afford the overhead for action information exchange, FP can be employed as the basic solution for many resource management games in wireless networks. In [140],

[8]This is equivalent to the condition when the best-response payoffs in the NE are strictly greater than the other possible payoffs for all the agents.

an FP-based multi-agent learning algorithm is employed by the secondary nodes in an ad-hoc DSA network to learn the strategies for forwarding delay-sensitive packets. In [140] the condition of channel availability is characterized by the matrix of spectrum opportunity, and the condition of channel contention is characterized by the interference matrix from both the PUs and the rival SUs. With the learning scheme proposed in [140], each SU needs to collect the information about the spectrum opportunity matrix locally, and establish its local interference matrix according to the action information collected from its neighbors. Then, every SU tracks the frequency of action selection by its neighbors according to a modified version of (26) with a discount factor κ_i^{t-1}. Each SU also needs to determine a subset of feasible actions that do not interfere with higher priority traffic. This is done through estimating the expected interference based on the policy estimation model in (27). The local deterministic best response is calculated based on minimizing the expected effective transmission time over the candidate links.

Another example of FP can be found in [141], which applies FP to obtain a defense mechanism against eavesdropping and jamming attacks in the uplink of a cellular network consisting of multiple relays (Figure 13). In the defense-attack game, the normal/malicious nodes are assumed to be able to observe the actions of other nodes, so they can use the models in (26) and (27) to estimate the other nodes' policies. Instead of directly obtaining a deterministic strategy based on the local best response, each normal node updates its mixed strategy at time slot t as follows [71]:

$$\pi_i^t(m) = \pi_i^{t-1}(m) + \frac{1}{t}(I(a_i^t, m) - \pi_i^{t-1}(m)), \quad (28)$$

in which m is the index of the candidate relays. The malicious node adopts a similar policy-updating rule based on its own action set for attacking. The actions of each node at round t are selected from the best response based on the expected private utility with the locally estimated policy vector (π_i^t, θ_i^t). The same learning rule as in [141] can be found in [142], which uses the local policy updating rule in (28) to learn the strategy in a continuous strategy space for power allocation. In [142], such a learning scheme is referred to as the *best response dynamics* of the power allocation game, and is proved to be able to converge to the ϵ-equilibria. Such a learning rule is also adopted in [143], which formulates a hierarchical network formation game for nodes in a multi-hop wireless network to select relays. In [143] the relay selection game is decomposed into multi-layers and solved using a backward induction method from the sink to the source. The learning scheme defined by (26)-(28) is applied to each layer-game and the mixed strategies are obtained from the local best responses.

With the standard FP, local actions are updated based on the best responses, which are generally of pure strategies. As pointed out by [33], one drawback of such an FP-based learning scheme lies in the discontinuity of agent behaviors, for a small change in the opponent-policy estimation may result in an abrupt local-behavior change. Due to this, a Smoothed-FP (SFP) procedure was proposed through search-
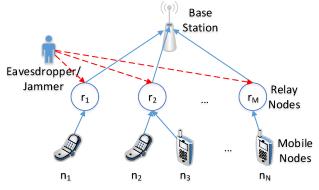

Fig. 13. A network consisting of M one-hop relays and N wireless users that is subject to eavesdropping/jamming from one active malicious node [141].

ing the best response with a modified local objective function that is perturbed by a differentiable, strictly concave function. Assume that the best response is obtained through maximizing a payoff function $u_i(\pi_i, \pi_{-i})$. Then the operation for obtaining the smoothed best response $\mathrm{BR}(\cdot)$ can be used to replace the original best response $\arg\max u_i(\pi_i, \pi_{-i})$:

$$\mathrm{BR}(\pi_{-i}) = \arg\max_{\pi_i} \{u_i(\pi_i, \pi_{-i}) + \nu\eta_i(\pi_i)\}, \quad (29)$$

in which the perturbation function η_i is typically given as the entropy function of π_i:

$$\eta_i(\pi_i) = \sum_{a_i \in \mathcal{A}_i} -\pi_i(a_i)\log\pi_i(a_i). \quad (30)$$

Problem (29) with (30) can be explicitly solved as:

$$\mathrm{BR}(\pi_{-i}) = \frac{\exp((1/\nu)u_i(a_i, \pi_{-i}))}{\sum_{a \in \mathcal{A}_i}(\exp(1/\nu)u_i(a, \pi_{-i}))}, \quad (31)$$

in which ν is the weight of the perturbation term that controls the strategy exploration rate. It has been proved that for any average-reward repeated game, we can always find the ν that makes the payoff of agent n under $\mathrm{BR}(\pi_{-n})$ to be sufficiently near the real best-response payoff (Proposition 4.5 of [33]). The SFP-based learning scheme is also known as the stochastic FP. Unlike standard FP, in SFP it is not necessary to observe the opponents' actions or even know the structure of the local utility functions. Instead, the expected payoff $u_i^t(a_i, \pi_{-i})$ in (31) is estimated based on local information as follows:

$$\tilde{u}_i^t(a_i) = \frac{1}{\kappa_i^{t-1}}I(a_i^t, a_i)\left(u_i^t(a_i) - \tilde{u}_i^{t-1}(a_i)\right) + \tilde{u}_i^{t-1}(a_i), \quad (32)$$

in which κ_n^t and $I(a_n^t, a_n)$ follow the same definitions as in (26), and $\tilde{u}_n^t(a_n)$ is the estimate of the expected utility $u_i^t(a_i, \pi_{-i})$. The local mixed policy is usually updated in the following form:

$$\pi_i^t(m) = \pi_i^{t-1}(m) + \alpha_t\left(\mathrm{BR}(\tilde{u}_i^t(a_i)) - \pi_i^{t-1}\right), \quad (33)$$

in which $\mathrm{BR}(\tilde{u}_i^t(a_i))$ is calculated based on (31) with the payoff estimated by (32) and α_t is a learning factor.

It is worth noting that with both value iteration in (32) and policy iteration in (33), SFP is usually considered as a typical form of CODIPAS-RL methods (see the example in [144]). Generally, the convergence conditions of SFP are based on the analysis of Lyapunov stability of the corresponding perturbed

best response dynamic [76]. A summary of these conditions for different types of games is given as follows[9]:

Theorem 3 (Convergence of FP [76]). *Consider SFP defined by (29)-(33) starting from an arbitrary strategy in game G.*

(i) *If G is a two-player symmetric game with an interior Evolutionary Stable Strategy (ESS) or a two-player zero-sum game, then SFP converges with probability one to an NE.*

(ii) *If G is an N-player potential game, then SFP converges in a subset of the rest points of the perturbed best response dynamic. If all the rest points of the perturbed best response dynamic are hyperbolic and two-order continuous, SFP converges to an NE with probability one.*

(iii) *If G is an N-player supermodular game, then SFP converges almost surely to a rest point of the perturbed best response dynamic. In particular, if the rest point is unique, then SFP converges to the NE with probability one.*

With the property of requiring no information exchange, SFP is considered an important tool in self-organized learning for resource allocation games. In [145], SFP is applied to the power control game in wireless ad-hoc networks. According to Theorem 3, SFP is guaranteed to converge to a stationary point (with a non-zero probability to an NE) for a supermodular/potential game. In order to take advantage of such a property, a supermodular utility function is designed for each node in [145], and the convergence with SFP is thus guaranteed. However, since the utility function in [145] is monotonically decreasing, the learning scheme will finally converge to the unique NE of that game, which corresponds to all users transmitting with zero power. This problem of utility function design is addressed in [116] by studying the power allocation problem in a small-cell network through a non-trivial Stackelberg game [18]. This game design is intended to balance the femtocell power efficiency and interference control in the macrocell. The supermodularity property is retained for the femto link utility, and the SFP-based scheme give in (31-33) is applied to the follower game among the femtocells. The same learning mechanism is adopted in [115], which considers the power allocation in the femtocells as a common-payoff game (thus a potential game). With the assumption of the common-payoff game, it is proved in [115] that the ϵ-equilibrium is guaranteed to be reached in the potential game by the SFP-based learning algorithm.

2) Gradient Play: Compared with FP, Gradient Play (GP) adjusts the strategy of one agent based on the gradient ascent dynamics instead of directly jumping to the best response based on the empirical frequencies of the opponent agents' action selection. Therefore, GP can be viewed as a "better response" algorithm. Mathematically, following the learning scheme of the standard GP, each agent in the repeated game updates its strategy on selecting a_i according to [71]:

$$\pi_i^{t+1}(a_i) = \left[q_i^t(a_i) + \epsilon_t(\nabla_{\pi_i} u_i(\pi_i, \theta_i^t(a_{-i}))) \right]^{\Pi_i}, \quad (34)$$

[9]About the definitions of ESS, rest point and supermodular game, please refer to [18] for more details.

where ϵ_t is the time-varying step size, $[\cdot]^{\Pi_i}$ defines the projection onto the strategy space Π_i of agent i, $\theta_i^t(a_{-i})$ is the estimated opponent-action frequency, which can be derived following (27), and $q_i^t(a_i)$ is the estimated local-action frequency, which can be derived in the same manner as (28):

$$q_i^{t+1}(a_i) = q_i^t(a_i) + \frac{1}{t+1}(I(a_i^t, a_i) - q_i^t(a_i)), \quad (35)$$

where action a_i^t is generated as random outcomes of the evolving strategies q_i^t. Following (34) and (35), the strategy of each agent is a (projected) combination of its own empirical action frequency and a gradient step based on the estimated opponents' action frequency. According to [71], [146], GP in continuous games is guaranteed to converge within a distance of order of ϵ_t of the NE of the game, if the NE is a strict one. However, GP cannot converge to a completely-mixed NE of the game (see Lemma 4.1 of [71]). Due to such a limitation on convergence condition, the basic form GP in (34) is rarely used directly in the solution to networking problems.

As an improvement to the basic form GP, Derivative-Action GP (DAGP) is developed in [71]. By introducing parameter $v_i^t(a_i)$ to approximate the first-order derivative of q_i, the updating mechanism of DAGP is defined as follows [146]:

$$v_i^{t+1}(a_i) = v_i^t(a_i) + \frac{\mu_t}{t+1}(q_i^t(a_i) - v_i^t(a_i)), \quad (36)$$

$$\pi_i^{t+1}(a_i) = \left[q_i^t(a_i) + \epsilon_t(\nabla_{\pi_i} u_i(\pi_i, \theta_i^t(a_{-i}))) + \mu_t(q_i^t(a_i) - v_i^t(a_i)) \right]^{\Pi_i}, \quad (37)$$

where q_i^t is updated following (35), $[\cdot]^{\Pi_i}$, ϵ_t and θ_i^t are obtained in the same way as in (34), and μ_t is a large factor satisfying $\mu_t > 0$. According to [71], [146], for large $\mu_t > 0$, if ϵ satisfies certain conditions (see Theorem 4.2 of [71] and Theorem 3.1 and Theorem 3.3 in [146] for more details), the strategy π_i^t is asymptotically locally stable and converges to the NE with a non-zero probability.

GP and DAGP not only require the agents to be able to track the frequency of both the local actions and the opponent actions, but also require that the structure of local utility functions is known to each agent. Compared with FP and SFP, the most important feature of the GP-based learning algorithms is that the updating mechanism can be easily extended to the cases of continuous games. In [147], standard GP is applied to the continuous, random medium access game, in which a set of wireless nodes learn to play the random access strategies p_i ($0 \leq p_i \leq 1$) after observing the vector of channel contention signal $\mathbf{q}_i$. Instead of directly adapting to the contention signal $\mathbf{q}_i$, each wireless node introduces a price function $C_i(\mathbf{q}_i)$ to adjust its local net payoff with the original utility function $U_i(p_i)$ as $u_i(\mathbf{p}) = U_i(p_i) - p_i C_i(\mathbf{q}_i)$. In [147], the random access game is proved to have a unique nontrivial NE (namely, $\nabla_{p_i} u(p_i^*, p_{-i}^*) = 0$ at the NE (p_i^*, p_{-i}^*)), and that the standard GP converges geometrically to the nontrivial NE if a certain condition is satisfied with the step size ϵ^t in (34). The application of standard GP can also be found in the power control game of a multi-cell CDMA network with dynamic handoffs between cells [148]. After introducing a pricing mechanism with the cost function based on the local power consumption, the game formulation in [148] adopts

a payoff function that is twice continuously differentiable, non-decreasing and strictly convex. It is proved in [148] that standard GP is able to exponentially converge to the smallest convex set which contains all the possible NE of the power control game, if the spreading factor of the CDMA system satisfies certain conditions[10].

One typical example of applying the DAGP-based learning to networking problems can be found in [149], which formulates the interference coordination problem in a multi-link MIMO system as a noncooperative game. In the game, the covariance matrix of the signal of each link is considered as the local strategy and is drawn from a common, continuous strategy space. The matrix form of (37) is adopted and guaranteed to converge to a unique NE of the game, if the covariance matrix of the total interference and noise at the receiver of each link satisfies a certain condition.

3) Learning Automata: As introduced in Section II-A, LA is featured by the process of action selection based on policy iteration using only local information. For non-game-based wireless networking problems, (distributed) LA has been shown to be efficient in the scenarios which can be formulated to be of single state and controlled by a single active decision-making entity at one time instance. Successful applications of LA in these scenarios can be found in the works such as multipath on-demand multicast routing in CRNs [150] and multicast routing in mobile ad-hoc networks [151]. When it comes to the more complicated framework of network control games, most of the LA-based learning schemes are employed to obtain NE policies. As a special case of the general LA updating rule (11), L_{R-I} learning has been widely applied to network control problems due to its simplicity and convergence property. By abusing the notations in (11), the rules of L_{R-I} learning can be expressed as (38):

$$\pi_i^{t+1}(a_i) = \begin{cases} \pi_i^t(a_i^t) + \mu \tilde{r}_i^t(1 - \pi_i^t(a_i^t)), & \text{if } a_i^t = a_i, \\ \pi_i^t(a_i) - \mu \tilde{r}_i^t \pi_i^t(a_i), & \text{if } a_i^t \neq a_i, \end{cases} \quad (38)$$

where μ $(0 < \mu < 1)$ is a learning parameter. The convergence property to the NE for the learning mechanism in (38) in a general noncooperative game has been proved in [152]:

Theorem 4. *In a repeated game $G = \langle \mathcal{N}, \mathcal{A} = \times \mathcal{A}_n, \{0 \leq \tilde{r}_n \leq 1\}_{n \in \mathcal{N}} \rangle$, with each agent employing L_{R-I} learning, the following statements are true if μ in (38) is sufficiently small:*

- *all stationary points that are not NE are unstable, and*
- *all strict NE in pure strategies are asymptotically stable.*

However, no uniform expression is provided in the literature to obtain the normalized environment response function $\tilde{r}_i^t$ in (38). For example, in [153], standard L_{R-I} learning is adopted to manage the opportunistic spectrum access by N SUs over M primary channels with a fixed transmit rate R_m on channel m. In this case, the normalized random reward $\tilde{r}_i^t$ is obtained as follows:

$$\tilde{r}_m^t = u_m^t / (\max_n R_n), \quad (39)$$

where u_m^t is the instantaneous reward of SU m after considering the PU activities and the channel contention with its rival nodes. The opportunistic spectrum access game is further modeled as an exact potential game. Therefore, at least one pure-strategy NE exists for the game [18]. According to Theorem 4, L_{R-I} learning ensures the convergence to the pure-strategy NE in the opportunistic spectrum access game. Apart from [153], the standard L_{R-I} learning scheme can be found as a frequent solution to the problems whenever the convergence property of Theorem 4 is satisfied and the existence of a pure-strategy NE can be proved. The applications of the standard L_{R-I} learning scheme range from relay-selection in the cooperative network [154] to the CSMA-based DSA management [155] and the MIMO-based DSA management [156] in the CRNs.

In contrast to the aforementioned works, the variation of the standard L_{R-I} learning mechanism using a different strategy-updating rule can also be found in the studies such as [157]. In [157], a discrete power control problem in a CDMA-like cellular network with mutual interference is modeled as a repeated noncooperative game. In the power control game, each node only knows its local payoff measured as the power efficiency. The modified linear-reward-inaction updating rule in [157] is defined as follows:

$$\pi_i^{t+1}(a_i) = \begin{cases} \pi_i^t(a_i^t) - \mu \tilde{r}_i^t \pi_i^t(a_i^t), & \text{if } a_i^t \neq a_i, \\ \pi_i^t(a_i) + \mu \tilde{r}_i^t \sum_{a \neq a_i} \pi_i^t(a), & \text{if } a_i^t = a_i. \end{cases} \quad (40)$$

Let u_i^t denote the utility of node i by choosing a discrete power level a_i^t for transmission at time t. Then, the normalized utility feedback $\tilde{r}_i^t$ is obtained as follows:

$$\tilde{r}_i^t = \frac{u_i^t - \min_i\{u_i\}}{\max_i\{u_i\} - \min_i\{u_i\}}. \quad (41)$$

The major difference between (40) and (38) lies in the way of updating the probability of choosing an action when the action results in a new reward. Under this learning algorithm, the evolution of the power selection becomes a Markov process. Following the same approach of proving the convergence property based on Ordinary Differential Equation (ODE) analysis and Lyapunov's stability theorem as in [152], it is proved in [157] that the LA-based learning scheme in (40) will only converge to the mixed-strategy NE of the considered power control game if the learning step μ is sufficiently small.

In addition to L_{R-I} learning, other learning schemes based on the general LA updating rule in (11) are also employed for resource allocation in the CRNs. In [158], an LA mechanism based on the softmax (Logit) function is applied to learn the ϵ-optimal solution to the traffic allocation problem in a multi-hop cognitive wireless mesh network. With the proposed LA mechanism, node i's local action to select link k for transmitting at the n-th possible rate is determined by the softmax function:

$$\pi_{i,k}^n = \frac{\exp(w_{i,k}^n)}{\sum_{m=0}^N \exp(w_{i,k}^m)}, \quad (42)$$

where N denotes the number of possible transmit rates and the intermediate parameter $w_{i,k}^n$ is updated according to the
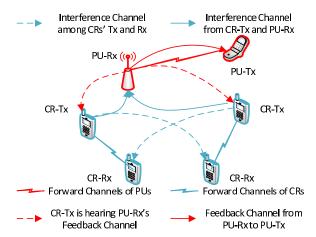
Fig. 14. A toy example of power allocation in the multi-user CRN with limited ability of acquiring the strategy information from other CRs [159].

following LA rules:

$$w_{i,k}^n(t+1) = \begin{cases} w_{i,k}^n(t) + \alpha_t \Xi(t)(1 - \frac{\exp(w_{i,k}^n)}{\sum_{s=0}^N \exp(w_{i,k}^m)}) \\ + \sqrt{\alpha_t}\xi_{i,k}^n(t), & \text{for } n = j; \\ w_{i,k}^n(t) + \sqrt{\alpha_t}\xi_{i,k}^n(t), & \text{for } n \neq j. \end{cases} \quad (43)$$

In (43), α_t $(0 < \alpha_t < 1)$ is the learning rate and $\xi_{i,k}^q(t)$ is obtained from a set of i.i.d. random variables with zero mean. $\Xi(t)$ is the normalized utility feedback that is provided by the gateway node. In order to ensure the convergence of the learning algorithm in (43), the traffic engineering game is modeled as a team game with the identical payoff (hence a potential game). Thus the SUs need to share the information on the global, normalized utility feedback $\Xi(t)$ for updating the value of $w_{i,k}^q(t)$. In [158], the value of $\Xi(t)$ is obtained from arbitrarily scaling the sum of the local payoff functions down to the range of $[0, 1]$. By allowing information exchange and constructing an N-person potential game, it is proved in [158] that for sufficiently small values of α_t and the variance of $\xi_{i,k}^q(t)$, the LA mechanism in (43) is guaranteed to achieve the ϵ-optimal solution to the traffic engineering problem.

In [159], Bush-Mosteller LA [70] is adopted for learning the NE of the repeated power control game in a CRN with the set of power constraints on the aggregated interference experienced by each PU (Figure 14). Bush-Mosteller learning, also known as the linear reward-penalty LA, can be viewed as a general form of L_{R-I} learning [160]. In [159], the CRN is assumed to be composed of N SUs and M PUs. The wireless channels are assumed to be stationary, and the SUs are able to monitor each PU's feedback indicating the sum of interference to each PU receiver. It is also assumed that no SU can observe the strategies of the other SUs (see Figure 14). Let $U_k(\pi_k, \pi_{-k})$ be the expected utility of SU link k and $W_l(\pi_k, \pi_{-k})$ be the corresponding expected interference at PU l, the constrained game is transformed into an unconstrained game with the help of the Lagrange multipliers. The Lagrange function of SU k is defined with a regularization term $\delta/2 \left(\|\pi_k\|^2 - \|\boldsymbol{\lambda}_k\| \right)$ as follows:

$$L_k^\delta(\pi_k, \pi_{-k}, \boldsymbol{\lambda}_k) = U_k(\pi_k, \pi_{-k}) - \sum_{l=1}^M \lambda_l(W_l(\pi_k, \pi_{-k}) - \overline{W}_l) - \frac{\delta}{2} \left(\|\pi_k\|^2 - \|\boldsymbol{\lambda}_k\|^2 \right), \quad (44)$$

where λ_l is the Lagrange multiplier for the constraint from

PU l, $\boldsymbol{\lambda}_k$ is the vector of λ_l and $\overline{W}_l$ is the maximum level of the interference to PU l. It is shown in [159] that finding the equilibrium point of the original constrained power control game is asymptotically equivalent to determining the equilibrium point of the unconstrained game with the regularized function given in (44). The following learning scheme, based on linear reward-penalty LA, is adopted to update the local policies:

$$\pi_k^{t+1} = \pi_k^t + \alpha_k^t [\mathbf{e}_{N_k}(P_k^t) - \pi_k^t + \frac{\tilde{r}_k^t(\mathbf{e}^{N_k} - N_k\mathbf{e}_{N_k}(P_k^t))}{N_k - 1}], \quad (45)$$

where P_k^t is the power level that SU k chooses at iteration t. $\mathbf{e}_{N_k}(P_k^t)$ and $\mathbf{e}^{N_k}$ are defined as follows:

$$\mathbf{e}_{N_k}(P_k^t) = (\underbrace{0, \ldots, 0, 1}_{i_k}, 0, \ldots, 0)^T, \quad (46)$$

$$\mathbf{e}^{N_k} = (1, \ldots, 1)^T. \quad (47)$$

The normalized utility feedback $\tilde{r}_k^t$ is obtained based on the Lagrangian with the expected utility and interference being replaced by the instantaneous payoff and interference in (44). With a user-defined normalization procedure, the value of $\tilde{r}_k^t$ is scaled within the interval $[0, 1]^{11}$. The time-varying correction (adaptation) factors α_k^t also belong to the unit segment. Meanwhile, the Lagrange multiplier is updated as:

$$\lambda_l^{t+1} = [\lambda_l^t - \alpha_\lambda^t \psi_l^t]_0^{\lambda_{l+1}^+}, \quad (48)$$

$$\psi_l^t = \delta^t \lambda_l^t - \eta_l^t + C_l, \quad (49)$$

where η_l is the instantaneous sum of interference at PU l and δ^t is the regularization factor in (44), and $[\cdot]_0^{\lambda_{l+1}^+}$ is a projection operator. The learning scheme defined by (45)-(49) ensures the convergence to the NE, provided that the sequences $\{\eta_l^t\}$ and $\{\delta^t\}$ satisfy certain properties (see Assumptions A1-A3 in [159]), and the power control game is diagonal concave [70]. Compared with L_{R-I} learning, Bush-Mosteller LA requires stricter condition for converging to the NE. This is a major reason for impeding Bush-Mosteller learning from being widely applied to the wireless resource allocation problems. Due to the requirement for the game to be diagonal concave, and because the original SINR-based utility does not naturally possess the property of diagonal concavity, the authors of [159] use an arbitrarily designed utility function to replace the real expected mutual-interference-based local utility in order to derive the proper payoff function for the constructed power control game.

4) No-Regret Learning: Usually, the terminology "no-regret learning" is used to refer to any learning algorithm that exhibits the property of no-regret when compared with the set of some designated strategies [72], [161]. Formally, for an infinitely repeated game $G = \langle \mathcal{N}, \mathcal{A} = \times \mathcal{A}_n, \{u_n\}_{n \in \mathcal{N}} \rangle$, and given the adversary (deterministic) strategy a_{-i}, the regret of agent i for playing strategy π_i instead of choosing strategy a_i can be defined as the difference in its payoff obtained from playing these strategies:

$$R_i(\pi_i, a_i | a_{-i}) = u_i(a_i, a_{-i}) - u_i(\pi_i, a_{-i}). \quad (50)$$

[11]For the detailed derivation of $\tilde{r}_k^t$, please refer to (31) and (32) in [70].

Let $\phi(\cdot)$ denote a modification mapping $\pi_i' = \phi(\pi_i)$, where $\pi_i'(a) = \sum_{b:\phi(b)=a} \pi_i(b)$ $(a, b \in \mathcal{A}_i)$. Then, for a sequence of adversary strategies $\{a_{-i}^t\}$, we can define a general no-regret learning algorithm (also known as ϕ-no-regret learning) for agent i as follows [161]:

Definition 6 (ϕ-no-regret learning). *For a finite subset Φ of memoryless mapping ϕ, a learning algorithm that generates π_i^t is said to exhibit ϕ-no-regret if the regret of that learning algorithm,*

$$R_{i,\phi}(\pi_i^t, \phi(\pi_i^t)|a_{-i}^t) = u_i(\phi(\pi_i^t), a_{-i}^t) - u_i(\pi_i^t, a_{-i}^t), \quad (51)$$

satisfies the following condition:

$$D_i = \lim_{T \to 0} \sup \frac{1}{T} \sum_{t=1}^{T} R_{i,\phi}(\pi_i^t, \phi(\pi_i^t)|a_{-i}^t) = 0. \quad (52)$$

There are two well-studied categories of the ϕ-no-regret properties: no-external-regret and no-internal-regret [161]. The no-external-regret property is to minimize the regret with respect to any comparison class of algorithms that lead to deterministic strategies. In other words, for no-external-regret learning, the mapping $\phi(\cdot)$ satisfies $\phi(\pi_i) = a$ $(a \in \mathcal{A}_i)$. The no-internal-regret property is also known as no-swap-regret since the property of internal regret swaps the current online strategies as follows:

$$\phi_{a,b}(\pi_i(c)) = \begin{cases} \pi_i(c), & \text{if } c \neq a, b, \\ 0, & \text{if } c = a, \\ \pi_i(a) + \pi_i(b), & \text{if } c = b. \end{cases} \quad (53)$$

One well-known example for applying no-external-regret learning to the wireless networking problems is [162], which uses the random weighted majority (i.e., Hedge) algorithm [163] for learning the NE strategies in a channel allocation game in a CRN. With a careful utility design, the channel-allocation game is proved to be an exact potential game. Let $u_i^t(a_i)$ denote the cumulated instantaneous payoff received by SU i given the sequence of the adversary strategy $\{a_{-i}^t\}$, the mixed policy of SU i is updated as follows:

$$\pi_i^{t+1}(a_i) = \frac{(1+\mu)^{u_i^t(a_i)}}{\sum_{a_i' \in \mathcal{A}_i}(1+\mu)^{u_i^t(a_i')}}, \quad (54)$$

where $\mu > 0$. It is well-known that the learning scheme in (54) has a regret bound as $D_i^T \leq \mu/2$ [78]. Compared with the widely applied best-response-based learning schemes for potential games, which also ensure the convergence to the NE, the random weighted majority algorithm (54) does not need any information sharing between SUs.

The construction of a no-external-regret learning mechanism can be further illustrated by the example of [164], where the problem of collaborative sensing with malicious nodes in an N-channel CRN is studied. In the considered CRN, SU j is supposed to collaborate with a set of its neighbor SUs $\mathcal{N}_j$ and to choose whether to aggregate one of their sensing reports into its local channel-state prediction. At time t, a mixed policy $\pi_t^j = [\pi_{1,t}^j, \ldots, \pi_{|\mathcal{N}_j|,t}^j]$ is adopted to choose the reports from the SUs in $\mathcal{N}_j$. With the goal of minimizing the long-term expected loss due to false decision by choosing the sequence

π_t instead of the pure-strategy best response (internal regret), we have

$$\min_{\{\pi_t^j\}_{t=1}^T} \sum_{t=1}^{T} \left(\bar{l}^j(\pi_t^j, s^t) - l^j(j', s^t) \right), \forall j' \in \mathcal{N}_j, \quad (55)$$

where $l^j(j', s^t)$ is the instantaneous loss due to adopting the report by SU j', and $\bar{l}^j(\pi_t^j, s^t)$ is the average loss with policy π_t^j at channel state s^t: $\bar{l}^j(\pi_t^j, s^t) = \sum_{j' \in \mathcal{N}_j \cup \{j\}} \pi_{j',t}^j l^j(j', s^t)$. In [164], such a decision process is modeled as a two-player constant-sum game. In the game, SU j plays against nature[12], which plays as an adversary player and chooses state s aiming at causing the worst cost to SU j. The strategy-updating mechanism is designed upon the softmax function (42) with the accumulated instantaneous loss $\sum_{\tau=1}^{t} l^j(j'^{\tau}, s^\tau)$ being the argument of the logarithmic function $\exp(\cdot)$. It is shown in [164] that no-regret learning based on the softmax function converges to the NE, which is equivalent to the minimax value of the game.

Another category of no-regret learning algorithms that are widely applied in the context of network control aims at minimizing the internal regret and learning the CE in repeated games [72]. For a general repeated game $G = \langle \mathcal{N}, \mathcal{A} = \times \mathcal{A}_n, \{u_n\}_{n \in \mathcal{N}} \rangle$, the estimated average loss for agent i to play action a_i^t instead of playing a_i' at time t is given by:

$$D_i^t(a_i^t, a_i') = \frac{1}{t} \sum_{\tau \leq t} \left(u_i^t(a_i', a_{-i}^t) - u_i^t(a_i^t, a_{-i}^t) \right). \quad (56)$$

Based on (56), the regret of agent i for not playing a_i' is

$$R_i^t(a_i^t, a_i') = \max \left\{ D_i^t(a_i^t, a_i'), 0 \right\}. \quad (57)$$

With (57), the mixed policy of agent i is updated by

$$\pi_i^{t+1}(a) = \begin{cases} \frac{1}{\mu} R_i^t(a_i^t, a), & \forall a \neq a_i^t, \\ 1 - \sum_{a' \neq a_i^t} \pi_i^{t+1}(a'), & a = a_i^t, \end{cases} \quad (58)$$

where μ is a sufficiently large constant to ensure that π_i $(i \in \mathcal{N})$ is a well-defined probability.

Like the random weighted majority algorithm, the learning scheme defined by (56)-(58) to learn the CE does not need the agents to exchange the action/utility information. The no-internal-regret learning scheme ensures the asymptotic convergence to the set of the CE, according to Theorem 5 [72]:

Theorem 5. *If every agent plays according to the learning scheme defined by (56)-(58), the empirical distribution of the joint action selection:*

$$z_T(\mathbf{a}) = \frac{1}{T} |t \leq T : \mathbf{a}^t = \mathbf{a}| \quad (59)$$

converges almost surely to the set of CE of the game G as $T \to \infty$.

The applications of the learning scheme given by (56)-(58) to network control problems can be found in [165]–[168]. As one of the earliest works that employ no-regret learning in the network control problem, it is aimed at obtaining the

---

[12] The definition of nature in an extensive form game can be found in [18].

CE in a dynamic spectrum access game with an overlay CR network in [165]. No-regret learning is used for the SUs to address the problem of channel contention. It is shown that the performance at the CE obtained through learning is almost as good as the optimal equilibrium in the set of CE. In [166], a joint power-channel selection problem is studied in an underlay CRN with a free band and a set of price-charging PU channels. The no-regret learning algorithm (56)-(58) is aggregated with an auction game, which considers the SINR to the PU or the allocation power as an item for auction. The joint power-channel selection game is played in two levels. In the lower-level subgame, the SUs perform the SINR/power bidding game with a fixed set of PU-channel selection. In the higher-level subgame, the SUs adopt the no-regret learning algorithm (56)-(58) to obtain the CE in the channel-selection game. In [167], the learning scheme of (56)-(58) is adopted to obtain the CE strategies in a spectrum sensing game among heterogeneous SUs in an overlay CRN. In the game, each SU chooses either to cooperatively sense the PU channel that it is assigned to with some power consumption (i.e., with some cost), or to directly access the channel as a free rider (i.e., without any cost) based on the sensing reports by the neighbor SUs. With the proposed no-regret learning scheme, the strategies are obtained based on minimizing the total regret of the neighborhood set of an SU rather than the individual regret. It is shown in [167] that the learning scheme with the neighborhood regret can significantly outperform the learning algorithm based on the local regret. This is also considered as the main reason that motivates local SUs to share their local action and payoff information for neighborhood learning. In [168], the scheme in (56)-(58) is applied to learn the CE of the subcarrier allocation strategies in a multi-cell OFDMA network. Again, each link in the subcarrier allocation game does not need to know the private strategies and utilities of the other links.

The no-internal-regret learning scheme (56)-(58) only requires that the structure of the local payoff function is known to each agent. Compared with the NE-driven learning methods such as FP and best-response learning, no-internal-regret learning could achieve a better social performance (i.e., in terms of sum of the players' rewards). Since the set of CE is a convex polytope with all the NE lying on one of its sections [169], it is possible for the no-internal-regret learning algorithm to reach a CE that is not in the polygon of the NE, thus resulting in a better performance than any NE. Although the learning rule of (56)-(58) does not guarantee convergence to the social optimal CE, a number of empirical studies (e.g., no-regret learning in the cognitive congestion control games [170], [171]) show that the no-regret learning scheme can significantly outperform best-response learning and FP [170]. Moreover, its convergent strategy can be considered as a good approximation of the global optimal solution [171]. As a result, many studies consider the no-internal-regret learning scheme as an approach to implicitly enforce cooperation within the framework of general-sum noncooperative games.

B. Applications of Learning in the Context of Stochastic Games (SGs)

SGs generalizes both the repeated games and the MDPs by allowing the payoff of the players at each round of the game to be dependent on the state variable, whose evolution is influenced by the joint actions of the players. Compared with the models based on repeated games, SGs are considered a more practical tool for modeling the agent interaction in a stochastic wireless environment, especially when the elements of the wireless environment (e.g., channel states, buffer states and collision states) evolve stochastically and are influenced by the transmission strategies of the wireless agents. In the context of SGs, the model-free learning schemes are referred to the value/policy-iteration algorithms (e.g., the algorithms summarized in [172]) that do not require any a-priori knowledge about the state transition of the wireless system. We note that such a property makes model-free learning especially appropriate for finding the solution to the equilibria of the SGs in the context of wireless networks. This is because in most of the practical scenarios it is difficult to obtain all the details of the system dynamics due to the complexity of the network. In what follows, we organize our survey on learning in SGs according to the approaches used for experience updating (i.e., value-iteration-based learning vs. non-value-iteration-based learning).

1) Value-Iteration-Based Learning: In contrast to those model-based solutions which use linear programming to obtain the NE (see the example of a constrained power control SG [173]), value-iteration-based learning algorithms generally need to construct a series of intermediate "matrix games" from the original SGs. Consider a general discounted-reward SG, $G = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \{u_n\}_{n \in \mathcal{N}}, \Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$, a matrix game is defined based on the current estimation of the state value of the SG, which is derived in a similar way as (4):

Definition 7 (Matrix game [67]). *An n-player matrix game (also known as stage game) in an SG is defined as a tuple $G(\mathbf{s}) = \langle \mathcal{N}, \mathcal{A}_1, \ldots, \mathcal{A}_{|\mathcal{N}|}, \hat{Q}^t_{\beta,1}, \ldots, \hat{Q}^t_{\beta,|\mathcal{N}|} \rangle$, in which $\hat{Q}^t_{\beta,i}$ ($1 \le i \le |\mathcal{N}|$) is given by:*

$$\hat{Q}^t_{\beta,i}(\mathbf{s}, \mathbf{a}) = u(\mathbf{s}, \mathbf{a}) + \beta \sum_{\mathbf{s}' \in \mathcal{S}} \Pr(\mathbf{s}'|\mathbf{s}, \pi) V^\pi_{\beta,i}(\mathbf{s}'|\mathbf{s}, \pi_i, \pi_{-i}). \quad (60)$$

We note that in (60), $V^\pi_{\beta,i}(\mathbf{s}'|\mathbf{s}, \pi_i, \pi_{-i}) = E_\pi\{\hat{Q}^t_{\beta,i}(\mathbf{s}, \mathbf{a})\}$. Under policy π, transition probability $\Pr(\mathbf{s}'|\mathbf{s}, \pi)$ can be expressed as follows:

$$\Pr(\mathbf{s}'|\mathbf{s}, \pi) = \sum_{a_1 \in \mathcal{A}_1} \cdots \sum_{a_{|\mathcal{N}|} \in \mathcal{A}_{|\mathcal{N}|}} \bigg(\Pr(\mathbf{s}'|\mathbf{s}, a_1, \ldots, a_{|\mathcal{N}|}) \\ \times \pi_1(\mathbf{s}, a_1) \cdots \pi_{|\mathcal{N}|}(\mathbf{s}, a_{|\mathcal{N}|}) \bigg).$$
$$(61)$$

According to Definition 7, a general form of strategy searching based on value iteration can be implemented as in Algorithm 1 [172]. In (62) of Algorithm 1, operator $\text{Eval}_\pi(\cdot)$ computes (estimates) the expected payoff in the NE of the matrix game. The equivalence between the NE of the matrix game and the NE of the discounted SG is given by Theorem 6.

Algorithm 1 Value-iteration-based learning algorithm.

Require: Initialize $V_{\beta,i}^t, \forall 1 \le i \le |\mathcal{N}|$ arbitrarily.

while convergence criterion is not met **do**

(a) For state $\mathbf{s}$ at round t, update the estimated value of $\hat{Q}_i^t(\mathbf{s}, \mathbf{a})$ of the matrix game.

(b) For state $\mathbf{s}$, update the expected state value of $V_{\beta,i}^t(\mathbf{s})$ after computing the (mixed) equilibrium strategy $(\pi_i(\mathbf{s}), \pi_{-i}(\mathbf{s}))$:

$$V_{\beta,i}^t(\mathbf{s}_t) \leftarrow \mathrm{Eval}_\pi(\hat{Q}_{\beta,i}(\mathbf{s}, \mathbf{a})). \qquad (62)$$

end while

Theorem 6 ([67]). *The following are equivalent:*

- π^* *is an equilibrium point in the discounted SG, G, with equilibrium payoffs $(V_{\beta,1}(\pi^*), \ldots, V_{\beta,|\mathcal{N}|}(\pi^*))$.*
- *For each $\mathbf{s} \in \mathcal{S}$, strategy $\pi^*(\mathbf{s})$ constitutes an equilibrium point in static matrix game $G(\mathbf{s})$ with equilibrium payoffs $(\mathrm{Eval}_{\pi^*}(\hat{Q}_{\beta,1}(\mathbf{s}, \mathbf{a})), \ldots, \mathrm{Eval}_{\pi^*}(\hat{Q}_{\beta,|\mathcal{N}|}(\mathbf{s}, \mathbf{a})))$. The value of $\hat{Q}_{\beta,i}(\mathbf{s}, \mathbf{a})$ is given by Definition 7.*

According to Theorem 6, Algorithm 1 can be considered a combination of a matrix-game solver and a value-iteration-based state value learner. It works as the general form of a set of model-free strategy-learning algorithms, which differ from each other only in the way of defining operator $\mathrm{Eval}_\pi(\cdot)$. In [64], operator $\mathrm{Eval}_\pi(\cdot)$ in value iteration is implemented by a minimax optimization process, and the Q-value of each learning agent is updated through a standard single-agent Q-learning process. Such a learning scheme is known as minimax-Q learning. Specifically, the learning mechanism can be expressed by

$$\begin{aligned} Q_{\beta,i}^{t+1}(\mathbf{s}_t, a_i^t, a_{-i}^t) &\leftarrow (1-\alpha_t) Q_{\beta,i}^t(\mathbf{s}_t, a_i^t, a_{-i}^t) + \\ &\alpha_t \left(u_i(\mathbf{s}_t, a_i^t, a_{-i}^t) + \beta V_{\beta,i}^t(\mathbf{s}_{t+1}) \right), \end{aligned} \qquad (63)$$

$$V_{\beta,i}^t(\mathbf{s}_t) = \max_{\pi(\mathbf{s}_t, a_i)} \min_{a_{-i}} \sum_{\mathbf{a} \in \mathcal{A}} Q_{\beta,i}^t(\mathbf{s}_t, a_i, a_{-i}) \pi(\mathbf{s}_t, a_i), \quad (64)$$

$$\pi^t(\mathbf{s}, a_i) = \arg\max_{\pi(\mathbf{s}, a_i)} \min_{a_{-i}} \sum_{a_i} Q_{\beta,i}^t(\mathbf{s}, a_i, a_{-i}) \pi(\mathbf{s}, a_i). \quad (65)$$

The solution to (65) is usually obtained through linear programming, which requires that the matrix game of the SG is of complete information. It is worth noting that (64) is an approximation of the exact state value, $V_{\beta,i}^t(\mathbf{s}_t) = \max_{\pi(\mathbf{s}_t, a_i)} \min_{\pi(\mathbf{s}_t, a_{-i})} \sum_{\mathbf{a} \in \mathcal{A}} Q_{\beta,i}^t(\mathbf{s}_t, \mathbf{a}) \pi(\mathbf{s}_t, \mathbf{a})$, which cannot be obtained directly since the local strategies are usually private information. Due to the approximation, the updating mechanism in (63)-(65), although proved to be effective by empirical studies [64], does not provide a strict condition for convergence to the NE.

Minimax-Q learning is usually adopted to solve the problems which can be described as a constant-sum (also known as strictly competitive) game. One typical category of its applications in wireless networks is strategy-learning in attack-defense problems, since such problems can usually be modeled as a two-player, zero-sum game with the group of normal nodes and the group of malicious nodes treated as two super players. In [174], a two-player zero-sum SG is adopted to model the anti-jamming process of a group of SUs in the
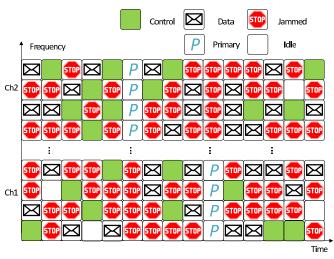

Fig. 15. A snapshot of the anti-jamming defense process in a multi-channel CRN (adapted from [174]).

CRN (Figure 15). Due to the random activities of the PUs, the channel-availability states viewed by the SUs are modeled as a group of independent, two-state Markov chains. In addition, for each channel, the channel quality measured by the local SNR is modeled as a finite state Markov chain. In [174], the devices in the CRN are divided into two groups: the normal SUs and jamming nodes. Both the normal SUs and the attackers access the PU channels in a slotted manner. At each time slot, the normal SUs will select a subset of channels for transmission while the attackers will select a subset of channels for jamming. The group of channels that are selected for transmission are further subdivided into control channels and data channels. For a normal SU, the non-zero gain of a channel can only be achieved when the channel is used for data transmission and at least one control channel selected by the normal SU is not jammed by the attackers. The goal of the normal SUs is to maximize the local channel utility. Based on the formulation of the two-player zero-sum SG, the standard minimax-Q-learning algorithm is applied for the normal SUs to find the equilibrium strategies in the stochastic attack-defense game. Convergence of the learning algorithm has been shown by empirical studies. Also, the numerical simulations show that minimax-Q learning outperforms both the myopic strategy, which does not consider the future payoff, and the fixed strategy, which uniformly selects the channels regardless of the attacker's strategy.

The application of minimax-Q learning in a similar scenario can be found in [175], which formulates the competition for open access spectrum in a tactical wireless network as a competitive mobile network game. The study in [175] extends the attack-defense model in [174] by dividing the competitive mobile network into two sub-networks: the ally network and the enemy network. Each network is composed of both communicating nodes and jamming nodes. The goal of the two networks is to achieve the maximum spectrum utility while jamming the opponent transmission as much as possible. The channel-availability state is jointly determined by the transmission-jamming actions of the two networks as a controlled Markov chain. Channel access in the competitive network is modeled as a two-player, zero-sum game, and

standard minimax-Q learning is adopted for both the ally and the enemy network to learn their equilibrium strategies. Apart from [175], other applications of minimax-Q learning can be found in [176], [177], which basically adopt the same framework of the two-player, zero-sum SG as in [174], [175] to obtain the anti-jamming scheme. In [176], minimax-Q learning in the SG is employed in a typical DSA network without considering the impact of jamming the control channels. In [177], the two-player SG model is extended to the scenarios of stochastic routing in a MANET, and the attack-proof strategy is obtained through minimax-Q learning.

For networking problems that need to be described as an n-player general-sum SG, a more general learning scheme can be implemented by replacing the minimax operator for $\text{Eval}_\pi(\cdot)$ with the operator that leads to the payoff of the NE in the general game. For the discounted-reward general-sum SGs, such a learning scheme is known as Nash Q-learning [66]. Nash Q-learning adopts the same Q-value updating scheme (63) as in the minimax-Q learning algorithm, and requires that the value of $V_{\beta,i}^t(\mathbf{s}_t)$ is obtained based on the matrix game NE of the SG. According to Theorem 6, as long as the NE of each matrix game obtained from the SG in stage $\mathbf{s}_t$ is used in (63) to compute the value of $V_{\beta,i}^t(\mathbf{s}_t)$, the learning process converges to the NE of the SG. For Nash Q-learning, operator $\text{Eval}_\pi(\cdot)$ can be expressed by:

$$V_{\beta,i}^t(\mathbf{s}_t) = \sum_{a_1 \in \mathcal{A}_1} \cdots \sum_{a_{|\mathcal{N}|} \in \mathcal{A}_{|\mathcal{N}|}} \prod_{i=1}^{|\mathcal{N}|} \pi_i^*(\mathbf{s}_t, a_i) Q_{\beta,i}^t(\mathbf{s}_t, \mathbf{a}). \quad (66)$$

In (66), $\pi_i^*(\mathbf{s})$ is the NE strategy of the matrix game at stage t when the payoff matrix of agent i is $Q_{\beta,i}^t(\mathbf{s}, \mathbf{a})$.

Theorem 6 also holds when the SG is based on average reward. The counterpart to Nash Q-learning in an average-reward SG is known as Nash R-learning [67]. Nash R-learning adopts the R-learning-based scheme for state-action updating as in (8) and (9), which can be summarized by the following equations:

$$R_i^{t+1}(\mathbf{s}_t, \mathbf{a}_t) \leftarrow R_i^t(\mathbf{s}_t, \mathbf{a}_t) + \\ \alpha_t \big(u_i(\mathbf{s}_t, \mathbf{a}_t) + V_i^t(\mathbf{s}_{t+1}) - R_i^t(\mathbf{s}_t, \mathbf{a}_t) - h_i^t(\mathbf{s}_t, \mathbf{a}_t) \big), \quad (67)$$

$$h_i^{t+1}(\mathbf{s}_t, \mathbf{a}_t) = h_i^t(\mathbf{s}_t, \mathbf{a}_t) + \theta_t V_i^t(\mathbf{s}_{t+1}), \quad (68)$$

where $V_i^t(\mathbf{s})$ is the equilibrium payoff of the stage game and is computed following (66).

When the goal of the learning process is to find the CE of the discounted-reward SG instead of the NE, Correlated-Q (CE-Q) Learning can be implemented based on the updating mechanism in (63)-(65) with the state value $V_{\beta,i}^t(\mathbf{s}_t)$ estimated at the CE strategies [68]. The equivalence between the CE of the original SG and the CE of the matrix game in each state still holds. Based on Definition 3 and Theorem 6, we have

Theorem 7 (CE in the SG [68]). *For a discounted-reward SG G, a stationary policy π is a correlated equilibrium if $\forall i \in \mathcal{N}, \forall \mathbf{s} \in \mathcal{S}, \forall \mathbf{a} \in \mathcal{A}$ with $\pi_i(a_i) > 0$, for all $a_i' \in \mathcal{A}_i(\mathbf{s})$*

$$\sum_{a_{-i} \in \mathcal{A}_{-i}} \pi(\mathbf{s}, a_{-i}|a_i) Q_{\beta,i}(\mathbf{s}, (a_{-i}, a_i)) \geq \\ \sum_{a_{-i} \in \mathcal{A}_{-i}} \pi(\mathbf{s}, a_{-i}|a_i) Q_{\beta,i}(\mathbf{s}, (a_{-i}, a_i')), \quad (69)$$

which defines the CE of the matrix game in $\mathbf{s}$ as $\pi(\mathbf{s})$.

For both the NE based Q-learning (Nash-Q and Nash-R) and the CE-based Q-learning (CE-Q), it is not specified how the equilibrium strategies $\pi_i^*(\mathbf{s})$ for each matrix game is obtained during the learning process. Since it is necessary for the game to be of complete information in order to immediately obtain the NE/CE of the matrix game, it is required that the learning agents should keep track of the entire Q-table from all the other agents at state $\mathbf{s}$ in order to compute the exact stage-game equilibrium. In practice, exchanging such information will result in a large transmission overhead, which is usually unaffordable in a wireless network. As a result, most of the existing studies apply heuristic methods to approximate the matrix game equilibrium. One example of payoff approximation at the NE of the matrix game can be found in [178], which decouples the wireless network into a group of Service Providers (SPs) and a single entity called Network Operators (NOs) for network virtualization. Each SP is responsible for reallocating the available spectrum resources to a group of end users, while the NO is responsible for allocating the time-varying spectrum resources to the SPs. Here, resource allocation through the interface between the NO and the SPs at each time slot is treated as an auction game with the NO acting as the auctioneer and the SPs acting as the bidders. The auction is performed following the Vickrey-Clarke-Groves (VCG) mechanism [18]. The entire auction process in the stochastic environment is modeled as a discounted general-sum SG, in which the channel state and the traffic state are assumed to be Markovian and the SP action is the selection of value functions through choosing the transmit rate. In [178], the matrix games of the original SG is referred to as the "current games". Also, to avoid directly computing the value of $V_{\beta,i}(\mathbf{s})$ in (66), a conjecture price which approximates the unit-rate price (strategy) of the NO in the future is introduced. A Q-value updating scheme which is analogous to the SAS-based Q-learning scheme is proposed, and the value of the conjecture price is updated using the subgradient method.

For networking problems which do not possess the single-server-distributed-agents property as stochastic auction games, the equilibrium strategies can be learned by implementing an appropriate amount of local information exchange. In [179], the problem of traffic offloading in a stochastic heterogeneous cellular network is first formulated as a centralized discrete-time MDP and then as an SG. In the SG, a group of macrocell BSs try to offload their downlink traffic to their corresponding group of small-cell BSs, which operate in the open access mode and share the same band with the macro BSs. Before the learning mechanism is implemented, the authors in [179] employ a standard state abstraction procedure based on linear state-value combination (see our discussion in Section III-A). The Q-values (i.e., the payoff of matrix games) are updated with the gradient-ascending method based on the gradient of the new Q-values after state abstraction. The matrix game in a given state $\mathbf{s}$ is modeled as a "virtual game" with common payoff by allowing the macro BSs to share their instantaneous spectrum utility with each other. Also, the action of each BS is updated using ϵ-exploration instead of directly computing

the mixed strategy of the matrix game. It is proved in that convergence (which may not be the NE) is gua[r] with probability one.

A different approach to approximate the matrix game librium with only local information in the SG can be in [180], [181], which employ the learning methods f[or] repeated games to learn the matrix game equilibrium str[a] and then use these intermediate strategies to appro[x] the state value $V_{\beta,i}^{\pi^*}(\mathbf{s})$ of the original SG. In [180] interference mitigation problem with a finite action discrete powers for both the PUs and the SUs in a C modeled as a discounted-reward SG. In [181], the cross resource allocation problem for layered video transm[] in a CRN is modeled as a discounted-reward SG. I[n] works, the goal of strategy learning is to find the CE o[f] respective SG. Both works treat the matrix game at state as a repeated game and adopt the no-internal-regret le[arning] method defined by(56)-(58) to approximate the CE st[] $\pi_i^*(\mathbf{s})$ at state $\mathbf{s}$. Let $\tilde{\pi}_i(\mathbf{s})$ define the intermediate st[] that is obtained with (58). Since with the no-internal[-regret] learning scheme, no action/payoff information excha[nge] needed, the strategy estimation in the SG is solely bas[ed] local information. The same method as in (63) is adopt[ed] Q-value updating, for which state value $V_{\beta,i}^{\pi^*}(\mathbf{s})$ under t[] strategy can be estimated as the expected payoff of the [] game:

$$V_{\beta,i}^t(\mathbf{s}_t) = \sum_{a_i \in \mathcal{A}_i} \tilde{\pi}^t(\mathbf{s}_t, \mathbf{a}) Q_{\beta,i}^t(\mathbf{s}_t, \mathbf{a}).$$

To further reduce the information-exchange overhead, the values of $\tilde{\pi}^t(\mathbf{s}_t, \mathbf{a})$ and $Q_{\beta,i}^t(\mathbf{s}_t, \mathbf{a})$ can be replaced by the conditional local strategy (given the adversary actions) and the Q-table based on the local state-action pairs [181], repectively. Such a two-fold, approximate learning scheme does not require the information exchange between wireless devices. However, compared with the original learning scheme in Algorithm 1, such a learning algorithm may suffer from using the non-CE policies in the matrix game and from the inaccurate estimation of $V_{\beta,i}^t(\mathbf{s}_t)$. Although empirical studies show that convergence can be achieved by the two-fold learning scheme, no theoretical support is available to guarantee the convergence to the CE.

2) Conjecture-Based Learning: Consider the problem of unguaranteed convergence due to the inaccurate estimation of the equilibrium strategies in the matrix games with two-fold learning, the concept of "conjecture" [37] about one player's opponent policies is introduced in several recent studies [182]–[184]. In an SG, the conjecture of agent i can be defined as any belief function $c_i : \mathcal{S} \times \mathcal{A}_i \to \mathcal{C}$, in which $\mathcal{C}$ is the space of agent i's conjectures (e.g., about the opponents' policies and states). In the case of policy conjecture, we can define $c_i^t(\mathbf{s}, a_{-i})$ as the conjecture of opponent policy $\pi_{-i}(\mathbf{s})$ by agent i at time t. With only local information, the most widely accepted conjecture updating mechanism is

$$c_i^{t+1}(\mathbf{s}, a_{-i}) = c_i^t(\mathbf{s}, a_{-i}) + \omega_i^{\mathbf{s}}(\overline{\pi}(\mathbf{s}, a_i) - \pi_i^t(\mathbf{s}, a_i)), \quad (71)$$

where $\overline{\pi}_i^t(\mathbf{s}, a_i)$ is the so-called reference point and is assumed to be common knowledge to all the players. With (71), the
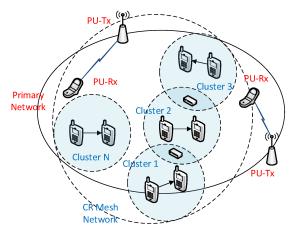

Fig. 16. Structure of underlay CR mesh network (adapted from [183]).

conjecture is used by local agent i to maximize its individual payoff in the condition of not knowing what the strategies of the other players are, or what their payoff functions are. (71) is obtained based upon the assumption that the other players will be able to observe player i's deviation from the reference point $\pi_i^t(\mathbf{s}, a_i)$, and in response to such a deviation, they will deviate from their own reference point by a quantity that is proportional to this deviation [37]. With conjecture $c_i(\mathbf{s}, a_{-i})$, the conjecture equilibrium can be defined as follows (extended from the definition in [182]):

Definition 8 (Conjecture equilibrium). *In the stochastic game G, a configuration of conjectures $\mathbf{c}$ and a joint policy π^* constitute a conjecture equilibrium if $\forall i \in \mathcal{N}$*

$$c_i^*(\mathbf{s}, \pi^*) = c_i(\mathbf{s}, \pi^*), \quad (72)$$
$$\pi_i^* = \arg\max_{\pi_i} Q_i(\mathbf{s}, \pi_i, c_i^*(\mathbf{s}, \pi_i)). \quad (73)$$

We take [183] as an example to explain the details of employing conjecture to learn in SGs. In [183], the power allocation problem in an underlay CR mesh network (Figure 16) is studied. The multi-node power allocation process is modeled as an SG, in which the local binary state of a secondary link is determined by the SINR level of its receiver. The local payoff is measured by the power efficiency. Compared with the standard matrix-game-based strategy-learning mechanism in (62)-(63), the authors in [183] constructs the Q-table with only local states and actions. Here, the policy conjecture is introduced to approximately learn the matrix game equilibrium strategy and the Q-value of the SG. Based on the conjecture-updating scheme in (71), the Q-value updating mechanism is defined as follows:

$$Q_{\beta,i}^{t+1}(s_i, a_i) = (1 - \alpha^t) Q_{\beta,i}^t(s_i, a_i) +$$
$$\alpha^t \left(\sum_{a_{-i} \in \mathcal{A}_{-i}} c_i^t(s_i, a_{-i}) u_i(s_i, a_i, a_{-i}) + \beta \max_{b_i \in \mathcal{A}_i} Q_{\beta,i}^t(s_i', b_i) \right). \quad (74)$$

The local policy π_i is updated using the Logit function (42). It is proved in [183] that the second term on the right-hand side of (74) is a contraction mapping operator and the learning scheme converges with sufficiently large number of iterations.

3) Other Learning Algorithms in SGs: For algorithms that do not work in the framework of hierarchical learning that is separated into learning in the matrix games and the original

---

Algorithm 2 Two-layer learning mechanism in the SG.

Require: Initialize $V_{\beta,i}^t$ and π_i^t, $\forall 1 \leq i \leq |\mathcal{N}|$.

    while convergence criterion is not met **do**

        Outer loop: $V_{\beta,i}^{t+1} \leftarrow \text{UpdateStateValue}(u_i^t, V_{\beta,i}^t, \pi_i^t, \pi_{-i}^t)$

        Inner loop: $\pi_i^{t+1} \leftarrow \text{UpdateStrategy}(V_{\beta,i}^t, \pi_i^t, \pi_{-i}^t)$.

    end while

SG, we simply refer them to the category of the "other learning algorithms". In these algorithms, the Q-learning-based value-iteration scheme for the payoff of the matrix game may not necessarily be applied, or the computation of the state value of the SG may not be needed. Due to the complexity of a general SG, most of the existing learning methods in this category cannot be represented by a single prototypical algorithm.

We note that for an SG, the property of the MDP generally requires that the state value of the game be computed following the Bellman optimality equation (in the general form as (3)), whenever a stationary policy is to be obtained. Extending from the value-iteration-based algorithm, we can construct a general learning scheme, which is composed of two learning loops: an inner loop that uses an appropriate scheme to approximate the SG equilibrium strategies π^* and an outer loop that employs an appropriate method to estimate the state value $V_{\beta,i}(\mathbf{s})$ of each player. Within this general framework, the construction of matrix games is not necessary. We can generalize the two-layer learning process in SG $G = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \{u_n\}_{n \in \mathcal{N}}, \text{Pr}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$ as Algorithm 2.

One widely-used two-layer approach for strategy learning in wireless SGs is to adopt FP-based policy updating as the inner-loop learning scheme. Such an approach of policy evolution can be found rooted in the model-based learning algorithms (namely, with known state-transition maps) [185]. Since the standard FP-based algorithm with (26) and (27) requires that each wireless node to track the opponent actions, extending FP-based learning from repeated games to the SG is considered a challenge due to the explosion of state-action dimensionality. In [186], such a challenge is resolved by regulating the SG into a sequential game, in which only one wireless node is allowed to update its action in each round. In [186], the problem of joint channel selection and power allocation for the SUs in an overlay DSA network is studied. With the assumption of a sequential game, each SU adopts a standard SAS-based Q-learning scheme as in (7) for updating the Q-table based on the local state-action pairs. To further reduce the state-action space, Q-learning is only applied to the strategy-learning for channel selection. The power adaptation is performed only after the channels are selected by the SUs. The FP-based mixed-strategy-updating scheme in [186] can be considered as a variation of the best-response-based strategy learning schemes described in (28).

It is also necessary to consider a different approach to update the state value for FP-based learning when the players in the SGs update their strategies simultaneously, because the state value of the MDP cannot be easily estimated by only tracking the opponents' actions. For those works that directly estimate the state value without using the TD-learning-based methods, it is also necessary to track the frequency of state transition in order to estimate the state transition probabilities. Examples of learning the state transition can be found in [87], [88]. In [87], secondary wireless stations compete with each other for network resources to transmit delay-sensitive in a stochastic CRN. In [88], a similar problem is specified in an overlay CRN with SUs competing for the vacant primary channels and determining transmitting parameters in a cross-layer manner. In both works, with the resource allocation problem in the CRN being modeled as SGs, it is required that the state transition frequencies of the opponents' local states are tracked by each SU. In order to reduce the information exchange overhead about local state transitions, an SU abstracts the state space by classifying the opponent SUs' state space purely based on its local observation. Instead of learning the real state-transition frequencies, the transitions of the abstracted state are recorded. The state value of the SG is updated based on the reduced states using the standard Bellman optimality equation (3).

The special structure of some SGs can also be exploited to simplify the learning process for the FP-based learning mechanism. One example of such exploitation can be found in [187], which models the distributed dynamic routing in multi-hop CRNs as an SG (Figure 17). Since the states of the routing SG in [187] are defined as the state of channel availability in the CRN, the SG is featured by the state transitions which only depend on the PU activities. The SUs in the network attempt to find the route for minimizing the packet-forwarding delay due to queueing and channel collision while keeping their interference to the PUs as small as possible. Since the delay over a path is equal to the accumulated delay caused by each link in the path, and the state transition is independent of the SU's actions, the original SG in [187] can be decomposed into a group of layered, stochastic subgames. Each subgame corresponds to a hierarchy level[13] in the routing path (see Figure 17). The structure (i.e., the payoff matrix) of each subgame can only be determined when the cost (measured in delay) of the next-layer game is determined. A backward induction method is adopted in [187] to compute the equilibrium payoff in the layered routing game. The computation starts from the subgame of the layer which ends at the sink SU to the subgame of the layer which begins from the source SU. Since the state transition is independent of the SU's actions, the stochastic subgame in each layer can be reduced to a group of repeated games with fixed states. Therefore, the learning of state value becomes unnecessary and FP-based learning guarantees the convergence to the global NE, as long as the routing costs at the equilibrium point of each subgame are properly propagated to their lower layers.

In addition to learning algorithms that follow Algorithm 2, a number of miscellaneous learning mechanisms are applied to SG-based problems in wireless networks. In order to reduce the requirement of information exchange or to achieve convergence, most of these learning mechanisms exploit special properties from the SG. As we have discussed in Section IV-A,

---

[13]According to [187], the hierarchy levels of the CRN are calculated along the "media axis", which is composed of a set of points. At these points, the lowest detection probability density of the PU's activities is (approximately) achieved.
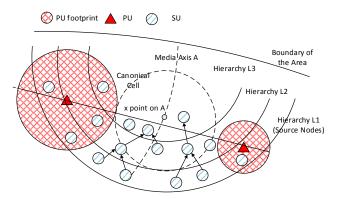
Fig. 17. A snapshot of a hierarchical multi-hop CRN under the PU interference footprint (adapted from [187]).

for the Aloha-like spectrum access problem in CRNs [126], the near-NE policies of the stochastic access game can be obtained if all the SUs update their local policies with the Logit function (42), and the Q-value at state $\mathbf{s}$ is updated following (23). In this specific scenario, the two-layer learning mechanism based on Q-value updating ensures the convergence to near-NE strategies of the SG without the need of any information exchange. In [188], [189], the structural property of a constrained SG is explored. Specifically, consider a utility-minimizing SG $G = \langle \mathcal{N}, \mathcal{S}, \times \mathcal{A}_i, \{c_i\}_{i \in \mathcal{N}}, \{d_i\}_{i \in \mathcal{N}}, \Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$ with c_i as the instantaneous local cost in the objective and d_i as the instantaneous local cost in the constraint. If the following assumptions are satisfied with G:

A1) the set of policies that satisfy the constraint of the SG is non-empty,

A2) the two cost functions c_i and d_i are multi-modular functions with respect to the actions and the state elements whose transition is a function of the joint local actions,

A3) the transition probability $\Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a})$ is submodular with respect to the actions and the state elements whose transition is a function of the joint local actions,

then G has the following property in the structure of the NE:

Theorem 8. *Assume A1-A3 hold, then the NE policy of each player i, π_i^*, is a randomized mixture of two pure policies: π_i^1 and π_i^2. Each pure policy is nondecreasing on the state elements whose transition is determined by the joint actions.*

Based on Theorem 8, the search for NE policies π_i^* can be reduced to finding a randomized mixture of discrete actions in the finite action set. A policy-iteration-based strategy-learning algorithm can be developed based on the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm [190]. In [188], the rate adaptation problem in a TDMA-based CRN is modeled as an SG with a latency constraint. In [189], the problem of joint source-channel rate adaptation in order to transmit layered video in a multi-user wireless local-area network is also formulated as an SG with the latency constraint. In both works, by showing that the assumptions A1-A3 hold in their respective SG-based model, the SPSA algorithm is applied for policy-learning. With the SPSA algorithm, no explicit state value learning is needed, and the local policies are updated with a gradient-based method with random policy perturbation. Given that the assumptions A1-A3 holds in the SG, the SPSA algorithm is proved to converge in distribution to the Kuhn

Tucker (KT) pair of the original constrained MDP (Theorem 3 in [188]).

In [79], another distributed learning algorithm is constructed based on the framework of L_{R-I} learning in the team SGs. A team SG can be considered as a variation of potential games when all the players in a SG share the same payoff function (i.e., fully-cooperative SG). With the proposed learning scheme, an LA is maintained for every state of the underlying Markov chain by each player in the SG. At any time instance, only one LA is activated by each player to learn its optimal action probabilities in the corresponding state. The introduction of LA reformulates the stochastic game between the $|\mathcal{N}|$ players into a repeated game between the $|\mathcal{N}| \times |\mathcal{S}|$ automata. Extending from the special case of team SGs, the convergence condition of the LA-based learning scheme for SGs is generalized by the following theorem:

Theorem 9 ([79]). *For SG $G = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \{u_i\}, \Pr(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \rangle$, assume that the multi-agent Markov chain corresponding to each joint policy, $\boldsymbol{\pi}(\mathbf{s})$, is ergodic. If $\boldsymbol{\pi}^*(\mathbf{s})$ is a pure NE policy in the view of $|\mathcal{N}|$ players in G, $\boldsymbol{\pi}^*(\mathbf{s})$ is also a pure equilibrium for the reformulated game between the $|\mathcal{N}| \times |\mathcal{S}|$ LA, and vice versa.*

According to Theorems 4 and 9, whenever an NE point in pure strategies exists in an SG (which is always the case for team SGs), the LA-based learning algorithm proposed in [79] is guaranteed to find the NE. However, it is worth noting that only maintaining an independent, repeated-game-based learning process (e.g., LA or SFP) for each state by the players may not necessarily produce the NE strategies for a general-case SG. Take the SFP learning scheme for example. In a general-case SG, the action-dependent state transition renders the Logit function in (31) no longer the solution to the perturbed best response. As a result, a Lyapunov function can not be found in the same way as for repeated games and the convergence property of the corresponding best-response dynamic in Robbins-Monro form is undermined. Therefore, special structure is required for the SGs if the repeated-game-based learning processes are to be adopted. In [191], a sufficient condition is given for the adoption of the CODIPAS-RL learning schemes (more specifically, LA and SFP-based learning) in the general-case two-player nonzero-sum SGs:

C1) the state transitions are independent of the player actions.

It is easy to prove that given condition C1, by fixing the state variable and solving for all the state-dependent NE with the repeated game-based learning algorithms discussed in Section V-A, we are able to obtain the state-independent NE of the two-player nonzero-sum SGs. The conclusion can be further extended to N-player games. When the state transitions are also independent of the current state, each player only needs to maintain a single learning process (see the examples in [191], [192]). However, due to the constraint on the state transition conditions, only a few applications of the SFP/GP/LA-based algorithms for the SGs-based network control problems can be found in the literature [192], [193].

VI. CHALLENGES AND OPEN ISSUES IN MODEL-FREE LEARNING FOR COGNITIVE RADIO NETWORKS

In this section, we expand our discussion to the challenges and open issues that are yet to be addressed in the area of learning for distributed control and/or wireless networking. In Section VI-A, different aspects of the learning mechanism goals are reviewed, and the potential conflict between these aspects is discussed. In Section VI-B, we propose a problem to cope with the outlier agents who do not (necessarily) follow a given learning rule in a learner set. In Section VI-C, the possibility of transferring experience from one learning scenario/process to a difference learning scenario/process is discussed. In Section VI-D, we discuss a problem on the coordination among simultaneous learning modules over different protocol layers for the same network entity.

A. The Goal of Learning: Self-Play, Stability and Optimality

Generally, the goal of a perfect self-organized learning mechanism for multi-agent decision making processes is to achieve self-play (autonomy), stability and optimality at the same time. However, it has been well-recognized that for multi-agent learning (more frequently in a stochastic scenario), improving system performance typically incurs more signaling and coordination, thus undermining the self-play structure. Especially, when learning is implemented under the framework of games, achieving any two goals of self-play, stability and network optimality is usually at the cost of undermining the third goal. In recent years, the relationship between the three parties of the goals in multi-agent learning has been discussed in many works, but mostly from a high-level theoretical perspective [26], [172], [194].

In regard to the applications of learning in wireless networks, the situations that have been discovered to keep consistence between a distributed solution and an optimal solution are limited within a small scope. One important case of these situations is the network control problems that is modeled as a potential game [77]. For potential games, the following properties [18] make it possible to achieve convergence to the optimal operation point through adopting the learning algorithms that we discussed in Section V-A:

- Every potential game has at least one pure strategy NE.
- Any global or local maxima of the potential function defined in the game constitutes a pure strategy NE.

Based on the above properties, it is only necessary to prove the uniqueness of the NE in a repeated game for learning processes to achieve optimal operation point with sequential best-response play [36] or no-regret learning. Apart from the works discussed in Section V-A, the applications of distributed learning in potential games in order to achieve global optimization can usually be found in a set of congestion-game-like problems such as [195], [196]. However, the potential game requires that local users are able to (implicitly) perceive the utilities of the entire network in order to establish the correspondence between the local utility function and the constructed potential function [77]. Since this requirement is at the cost of trading off the conditions for self-play, it

significantly limits the applications for the potential-game-based learning algorithms.

For other model-free distributed learning mechanisms in a multi-device wireless network, how to coordinate the goal of optimality and self-organization when adopting a learning scheme generally remains an open question. As a result, most current studies focus on ensuring convergence to the stable operation point in self-play by allowing a limited level of control signal exchange. Although there are a few already-known conditions that ensure the convergence of a learning algorithm, most of which are applicable to repeated games (e.g., Theorem 2 and 3), for most current studies, whether a stability condition can be found for a learning scheme also remains an open issue. In the literature, the approaches to find the convergence condition of the learning algorithms generally fall into two major categories. For learning processes that can be approximated with a linear system described as a set of ODEs in continuous time, the typical way of obtaining the convergence condition is to construct a Lyapunov function for the ODE-based dynamic and then prove that the strategy/utility updating mechanisms produce an asymptotic pseudo-trajectory of the flow defined by the ODE through the stochastic-approximation-based analysis (see the example in [73], [152]). The analysis of learning using the ODE-based approach can be found in [126], [142], [145], [159], [197]. For the situations which cannot be easily modeled as an linear, ODE-based system, the contraction-map-based analysis (see the example in [66]) can be considered as an alternative. Usually, the contraction map is considered appropriate for the analysis of SG-based learning when modeling the problem is of high complexity [183], [184]. Table XII summarizes convergence conditions for the multi-agent learning algorithms discussed in Sections III-V.

In addition to the issues associated to finding the convergence condition for a learning scheme, another concern when applying model-free learning in wireless networks is the convergence rate of learning algorithms. Although analytical results for the convergence rate of learning algorithms are highly desired, most of the existing studies are only able to show empirical results for the learning convergence rate through numerical simulations (see the examples in [88], [183]). The reason for this is partly due to the asymptotic convergence condition (if there is any), which requires for most of existing learning algorithms that the states and actions are visited infinitely to ensure the convergence. Given such a limitation, one known approach to analyze the convergence speed of a learning scheme is to view the learning process itself as a discrete time Markov chain. In this approach, the standard Markov chain analysis can be applied to obtain the expected time (number of iterations) to learn before reaching the chain's absorbing state (e.g., the equilibrium point of a repeated game). Such a technique can be found in the recent studies [198], [199]. In [198], the Markov-chain-based analysis is used to measure the lower bound of the iterations needed for the Logit-function-based learning scheme to leave a sub-optimal NE in a potential game for gateway selection [198]. In [199], the same method is employed to track the average iterations that a trial-and-error-based learning method

TABLE XII
A SUMMARY OF THEORETICAL CONVERGENCE CONDITIONS FOR THE MAS-BASED LEARNING ALGORITHMS

Problem Formulation Category	Learning Scheme	Convergence Condition	Stable Operation Point	Required Signaling
Loosely Coupled MAS	Distributed (independent) Q-learning	Generally not known	Sub-optimal	None
Repeated Games	Standard FP	Not guaranteed except in (a) two-player games and (b) multi-player game with common payoff [71]	ϵ-NE	Exchange of local-action information
	Stochastic FP	Not guaranteed except in (a) potential games, (b) supermodular games (c) two-player zero-sum games and (d) two-player symmetric games [76]	ϵ-NE	None
	Gradient play	Conditional convergence for strict NEs in multi-player games [71]	NE	Exchange of local-action information
	L_{R-I}	Conditional convergence for strict NEs in multi-player games (see Theorem 4)	ϵ-NE	None
	No-external-regret learning (Hedge)	Potential games	NE	None
	No-internal-regret learning	CE in multi-player games	Non-social-optimal CE [72]	None
Stochastic Games	Minimax Q-learning	Not known	NE	Knowing the structure of local payoff function
	Nash Q-learning/R-learning	Each matrix game has a unique NE [66], [67]	NE	Exchange of local action/payoff information
	Conjecture learning	Conditional convergence	Conjecture equilibrium	Knowing the reference point
	FP-based policy updating	Generally not known	NE	Exchange of local-action information

needs for reaching the NE of a joint channel-power selection game for the first time. However, such an approach could be computationally intractable when the system/learning scheme is too complicated, and it is yet to be found applicable to the more complex learning algorithms such as those in the SGs.

B. Heterogeneous Learning and Strategic Teaching in the Context of Games

For the existing studies of strategy learning in wireless networks, one most important assumption is that each individual agent abides by the same learning rule (or just uses variable parameters for the same learning scheme). Only with such an assumption, the convergence properties of the learning scheme can be mathematically tracked. However, in many practical scenarios, especially in the scenarios when malicious nodes exist in the network, such an assumption may not be applicable and the malicious nodes may intentionally deviate from the given learning rule. One possible scenario of such a case can be found in a selective-forwarding-based attack-defense game, in which a sophisticated attacker with the ability of selectively forwarding the received packets may wait and abide by the normal packet forwarding rule until some critical packets are sent to it before dropping. To the best of our knowledge, currently there are few (if not any) works discussing this situation.

To further demonstrate the situation in which a learner may benefit by deviating from a common learning rule, we introduce the concept of "strategic teaching", which is first discussed in the studies of economic games [200]. With strategic teaching, it is assumed that the game is composed of a number of adaptive players and sophisticated players. An adaptive player learns its strategy following the learning

scheme that it is assigned to. By contrast, the sophisticated players are able to adopt a non-myopically optimal strategy and afford a certain short-term loss. Since the adaptive learners will finally learn the best response to a pre-committed strategy by the sophisticated player under the given learning scheme, the sophisticated players will be able to induce the adaptive players to expect some specific patterns of strategies from them in the future [200]. Then, the sophisticated players will be able to take advantage of the behavior patterns that they "teach" the adaptive players. It has been found that a sufficiently patient strategic teacher can achieve as much utility as from first-play in a Stackelberg game[14] [200]. Thus, the sophisticated play may become a favorable way of strategy adoption for a noncooperative or a malicious node in the wireless network compared with the way of strictly following the same learning rule.

In [200], a heuristic, model-free learning method known as Experience-Weighted Attraction Learning (EWAL) [139] is applied to a repeated trust game (i.e., lender-borrower game) as the basis of both adaptive learning and sophisticated learning. In that game, M borrowers try to borrow money from each of a series of N lenders. A lender only makes a one-time binary decision on either *Loan* or *No Loan* in a single round out of a N-round game. A borrower makes a series of N binary decisions on *Repay* or *Default* regarding each lender that it borrows money from after observing the lender's decision. The sequences of the N-round stage-games (also known as supergames) are repeated for many times with a random order of lenders to make decisions with each sequence. In one sequence, one borrower is picked as the

---
[14]About the difference of a Stackelberg equilibrium and an NE, the readers are referred to [18] for more details.

common borrower in the game. All the lenders and some of the borrowers play as adaptive players and learn their strategies with EWAL. The rest of borrowers are assumed to be dishonest and adopt sophisticated play. It is assumed that the actions and instantaneous payoffs of one player are observable by the other players. For the adaptive players, EWAL uses the Logit-function-based rule as in (42) for strategy updating. Instead of directly using the instantaneous/accumulated payoff as the argument of operator $\exp(\cdot)$ in the Logit function, EWAL introduces the concept of experience accumulation through reinforcement and employs two new measurements to build local experience: the observation-equivalents of the past experience and the attraction to a specific strategy [139]. The former is similar to the action-frequency estimation in FP and the latter is used as the argument of the Logit function. In the game, the adaptive players apply EWAL twice to build their attraction first within a lending-borrowing sequence (i.e., supergame) and then across the consequent sequences. For the sophisticated borrowers, the learning process does not differentiate between attraction building within a supergame and across different supergames. A sophisticated borrower guesses how the lender learns according to the attraction value of the adaptive lender that it observes. Then, the policies of default and repay are sought by incorporating estimated policies of the lenders into the computation of its own sophisticated attraction function (see Section 4.1 of [200] for the details). It has been demonstrated in [200] that by adopting sophisticated play with the attraction updating mechanism based on lender policy estimation, the dishonest borrowers are able to outperform the adaptive borrowers which follow the same EWAL learning rule as the lenders. For simplicity, the mechanism of sophisticated play can be interpreted as playing additional tricks to the adaptive lenders by repaying frequently enough so if the dishonest borrowers do default, it won't lower the belief probability of the lenders about the trustworthiness of these borrowers below a critical level. Such an example provides an important insight into the possible strength of sophisticated play in repeated noncooperative games. However, few studies discuss such an issue in the context of wireless networks. Also, it is generally not clear how strategic teaching with sophisticated play in other forms can be enforced or avoided in the current framework of learning and in what ways it will affect the equilibria that can be reached.

C. Experience Transferring between Heterogeneous Learners

As we note from Sections III-V, one of the significant benefits of model-free learning is to allow the decision-making entities to learn the strategies from scratch without the a-priori knowledge of the wireless network. However, since model-free learning is based on trial-and-error, when the network environment has dramatically changed, the learners generally need to start the same learning process from the very beginning. One example of such scenarios can be found in interference mitigation problem for cellular networks, in which mobile stations may enter or leave the network frequently. For most of the existing model-free learning algorithms, such changes in the network topology mean the changes in the MDP

model of the network with new dimension of states/actions, if MDP-based learning is adopted, or the transition from an old network-control game to a new one since the set of players is different, if game-based learning is adopted. As a result, when it is required that the decision-making agents swiftly switch from an old scenario to a new one, the existing learning methods will face great challenges if they can only restart the learning process in the new scenario.

In order to address such a challenge, a natural consideration is to utilize the acquired experience of strategy taking which is obtained from the old scenario. We note that such a process is fundamentally different from the experience sharing process discussed in Section IV-B, since for the experience-sharing framework such as docitive networks, the parallel and homogeneous learning processes are assumed so the expert agent is able to share its better experience of the same stochastic process with the newcomers. In the scenarios of dramatical environmental changes, the experience transferring paradigm, Transfer Learning (TL) [138], is considered more appropriate for the tasks of sharing experiences of strategy taking between heterogeneous learning processes. Compared with the experience transferring between homogeneous learners, the motivation of TL is to transfer knowledge (i.e., experience) from the well-established learning processes (known as the source tasks) to the newly established learning processes (known as the target tasks) in a different situation. It is worth noting that under the framework of MDP-based learning, TL allows the difference in state spaces, state variables/transition, reward functions and/or sets of actions [138].

TL has been considered difficult to implement for learning in wireless networks. This is mainly due to the fact that it is difficult to find a proper mapping (either in value-function representation or directly in policy transferring [138]) to transfer between learning tasks with different action-state representations. For the applications in wireless networks, one example of policy-transferring TL can be found in [201]. In [201], a highly dynamic opportunistic network which is based on LTE-A is studied. The network topology is assumed to change with time, and the eNodeBs (eNBs) are supposed to be responsible for learning channel allocation under the conditions of mutual interference among the user equipments. The mechanism of policy transferring is adopted on the basis of two model-free learning algorithms: the linear reinforcement learning and the single-state Q-learning. The former employs a simple, linear updating function for state-value updating, while the latter applies Q-learning to update a state-less Q-table. For TL, one shot of the changing network topology is considered as a learning phase, then the objective of TL is to apply the experience learned in previous phases (sources) to the similar phases (targets) in the future. The eNBs which attempt to assign channels to the user devices for interference coordination work as the learning agents and obtain the spectrum priority through sorting the Q-table obtained in the current phase in a descent order. A policy function is designed to transfer the Q-table learned in a previous phase to the new phase through assigning weights to the source priority table to the target priority table in the new phase. Such a procedure of associating the channel priority in the target
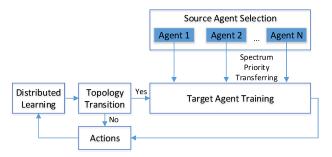
Fig. 18. Architecture of the policy-transfer mechanism in the LTE-A based opportunistic network [201].

Fig. 19. Architecture of the transfer-actor-critic algorithm [202].

task with the channel priority in the source target can be considered as initializing the learning process in the new phase with the transferring knowledge from the old phase. Thereby, the information from transfer learning and distributed learning is combined through weighting the values of channel priorities. The Q-table in the new phase is learned with the given reinforcement learning methods. The policy transferring process in [201] is demonstrated in Figure 18.

A different approach of applying TL to the wireless networking problems can be found in [202], where the authors apply TL to a series of actor-critic learning processes to coordinate BS switching/sleeping in a cellular network. In [202], the possibility of improper guidelines provided by transferred knowledge of the old task to the new task is considered. The actor-critic learning scheme is performed by a BS-operation controller, and is based on a multi-state MDP model for the traffic load of the serving BSs. Compared with [201], the difference of the TL mechanism in [202] lies in the way of adopting the transferred policies. Instead of using the static transferred knowledge for the initialization of the new learning phase, the experience in the new learning phase is divided into two sources: the "native policies" obtained through actor-critic learning and the "exotic policies" obtained as transferred policies from old tasks. The weight of the exotic policies contributing to the overall strategy selection decreases as the native learning process progresses. The learning-knowledge-transferring process is demonstrated in Figure 19. It is mathematically proved that regardless of the initial value of the overall policies and the transferred policies, the actor-critic-learning-based algorithm is guaranteed to converge. Also, numerical simulations show that TL does improve the learning speed when compared with the reinforcement learning methods without TL.

In the literature, most of the applications of TL in wireless networks are set in the scenarios which can be modeled as MDP-based MAS. With all the existing effort for establishing a general framework of applying TL to learning in wireless networks, the following questions are to be answered:

1) Whether and how can TL be applied between related games (e.g., symmetric games with the same structures of payoffs and actions, but with different sets of players) for accelerating the convergence speed to the equilibrium?
2) How can we measure the efficiency of knowledge transferring?
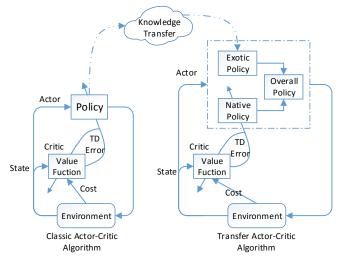3) Apart from policy transferring and value-function transferring, can TL also be applied to heterogeneous learning

processes using different learning schemes?

In the literature, few studies in wireless networks are found discussing the aforementioned topics. However, discussions on cross-game learning or cross-mechanism learning have already begun in the area of economic games [203] and automatic control [204]. Although the detailed discussion on these topics is beyond the scope of this survey, it is believed that addressing these issues will bring great improvement to the existing learning mechanisms in the CRNs.

D. Coordination of Learning Modules: Integration vs. Decomposition

In addition to the problems in heterogeneous learning processes, handling experience sharing or transferring knowledge among different network devices, the coordination of simultaneously learning modules may still be a challenging issue even within a single network device. As shown in our previous discussion, learning processes targeting at various functionalities (which may or may not involve the interactions with other users) can happen in any layer of the protocol stack (see Figures 7, 16 and 17 for example). Although many existing works have succeeded in applying the learning-based solution to their dedicated functionalities, a systematic discussion on coordinating these learning processes for different functionalities generally remain untouched in the current research progress. In the seminal work [205], it is pointed out that different functionalities across the protocol layers may exhibit a range of conflicts and/or dependence when working concurrently in the same network. Thereby, it becomes a natural idea to consider the solutions to the learning module coordination by first identifying the conflicts or dependence in practical scenarios.

Based on the work in [205], [206], we consider the following major conflicts and/or dependence among different network functionalities:

1) Logical dependence: this kind of dependence may arise when there is a logical dependence between the objectives of different network functions.
2) Parameter conflict/dependence: this kind of conflicts or dependence is triggered either when different networking

functions try to modify the same configuration parameters or when the parameters of one function depends on some other network parameters.

3) Measurement conflict: measurement conflicts exist if a learning module depends on the state of the other learning modules.

Logical dependence happens when different learning modules exhibit a hierarchical dependence on the output of each other. In this sense, the relationship between different learning modules in a CR device shares a lot of similarity with the relationship between the subtasks of a hierarchical reinforcement learning mechanism [207], [208]. The major difference is that in (MDP-based) hierarchical reinforcement learning, a single learning process is decomposed into a number of subtasks with their own sub-states, actions, transition functions and rewards in a top-down manner with the help of recursive value function decomposition[15] [208]. Since hierarchical learning requires to finish each child learning task before starting its parent task, it extends the MDP-based system model into a semi-MDP-based system model, in which the amount of time for the transition from one action to the next is a random variable due to the existence of the subtask sequences. By adopting the general idea of hierarchical learning, learning coordination with logical dependence can be considered as a reverse process of hierarchical learning by integrating the existing learning modules according to their dependence and forming a macro learning task. Practically, such an operation of module concatenation may be extended to the non-MDP-based learning mechanisms. For example, in [180], [181] a hybrid structure of both MDP-based Q-learning and repeated game-based no-regret learning is formed to approximate the equilibrium strategy of an SG. In those two cases, the expected utility based on the learned equilibrium of the repeated game can be considered as the instantaneous utility of the parent-level Q-learning process. However, the major difficulty in applying a hierarchical learning-based coordination mechanism lies in the uncertainty of convergence, as we have highlighted in V-B. Unlike the well-established examples of hierarchical learning in the domain of robot control [207], For the applications in CRNs there usually exists no terminal state for a subtask to determine when to stop its execution. As a result, when to start and terminate a task in the framework of hierarchical learning are usually determined empirically, and the convergence conditions of such a learning process still remains an open issue.

Unlike logical dependence, parameter conflict/dependence and measurement conflict are caused by the conflicts of the actions and states in different learning modules, respectively. For example, parameter conflict may happen between the inter-cell interference control and the coverage/capacity optimization modules of a cellular network. With respect to downlink transmit power control, the interference control module may want to decrease the transmit power in order to reduce the inter-cell interference, while the coverage/capacity optimiza-

tion modules may want to increase the transmit power to improve the local link quality at the same time. For those two kind of conflicts, one traditional solution is to build a decision tree to activate different decision modules according to the pre-determined conditions, which is also called trigger-condition-action points [205]. However, the trigger-condition-action based solution is a typical model-based method, and thus cannot be directly incorporated into the coordination process of learning modules.

Although no prototypical solution has been proposed to resolve conflicts 2) and 3), it is still possible to address these conflicts by imitating the existing model-based methods when some certain property can be found in the learning modules. Consider a general case where a number of learning modules share a subset of network states, and try to learn the strategy on the same action parameters to achieve different goals. To resolve the conflicts, we can adopt the idea of layering by decomposition in [16] to coordinate the learning modules. One typical way of doing so is to pick the objective of one network functionality as the major goal and treat the goals of all the other functionalities as constraints. It is worth noting that such an operation can be also considered as a way of integration. However, the ultimate goal of it is to create a structure of optimization which suits the further operation of decomposing it into interrelated but layered learning processes. A revisit to the work on layered Q-learning for video compression [93] helps to exemplify such an idea in details. In [93], a multimedia processing system considers three different concurrent objective functions, which are the video distortion at the codec level, the queueing delay for video frame processing in the pre-encoding buffer, and the energy cost in the OS/hardware layer. The distortion and queueing delay can be treated as two objective functions in the application layer of the system sharing the same system state, while the configuration that defines the energy cost (the operating frequency in this case) also determines the distortion of the compressed video. In [93], minimizing the queuing delay is considered as the main objective, and the rest two objective functions are treated as constraints. Conflicts between different functionalities can be easily found in this case, since increasing the operating frequency will lead to a better video quality but result in more energy consumption. By creating such a constrained optimization problem, a layered Q-learning mechanism is designed in a way that is similar to the procedure of dual decomposition. As briefly discussed in Section III, a two-layer learning framework is created in the following way. In the application layer, the Q-learning module receives the signaling from the OS/hardware layer about its action (frequency selection) information, and learns the local state value. In the OS/hardware layer, the local learning process receives the estimated Q-value of the application layer as part of its instantaneous utility, and then learns its own state value. Unlike the hierarchical learning based integration method, layered learning based on decomposition does not require that one learning process to be finished first before another learning process starts.

Like integration-based learning, the mathematical proof of convergence for decomposition-based learning is still

---

[15]A general principle for a hierarchical value function decomposition is that the reward function of a parent task is the state-value function of the child task [208].

rarely discussed in the existing literature. In the meanwhile, although considered more autonomous than the model-based coordination methods such as trigger-condition-action, decomposition-based learning needs a pre-determined constrained-optimization structure for layering of the learning processes. Such a requirement may limit the ability of decomposition-based learning in quickly responding to the requests of a certain network functionality that cannot be reached in the given constrained-optimization structure. From this point of view, finding a satisfying tradeoff between different functionalities still remains an open question for decomposition-based learning coordination.

VII. CONCLUSION

Owing to the distributive nature of cognitive wireless networks, model-free learning is especially appropriate for the wireless nodes to adaptively choose their transmission strategies in a self-organized manner without much requirement for knowing the network conditions. In this paper, we have provided a comprehensive survey on the applications of the state-of-the-art learning mechanisms in a wide range of scenarios of network modeling. With a broad-scope analysis and comparisons of the literature, we have focused on learning algorithms that can be categorized with a set of prototypical schemes. Briefly, these prototypical schemes includes MDP-based learning and experience sharing, conjecture-based learning, FP/GP-based learning, LA-based learning and no-regret learning. We have classified the various scenarios for the applications of learning into three major categories, namely, the SAS-based network control, the loosely-coupled MAS-based network control and the game-based network control. We have mainly focused on the following characteristics of the selected learning algorithms: (i) the ability of the learning schemes to achieve optimality/equilibria without knowing an a-priori model for the environment, (ii) the ability of the learning schemes to achieve optimality/equilibria without obtaining the information that is not locally available and (iii) the ability of the learning schemes to quickly adapt by exchanging experience. In addition to detailed reviews of the existing applications of learning in wireless networks, we have also discussed a variety of open issues that need to be addressed in future research. We hope this survey will serve as an important guideline for future research directions to further understand model-free learning mechanisms and expand their applications in cognitive wireless networks.

REFERENCES

[1] J. Mitola, "Cognitive Radio — An Integrated Agent Architecture for Software Defined Radio," DTech thesis, Royal Institute of Technology (KTH), Kista, Sweden, May 2000.

[2] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[3] B. Wang and K. Liu, "Advances in cognitive radio networks: A survey," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 1, pp. 5–23, Feb. 2011.

[4] S. Haykin, *Cognitive Dynamic Systems: Perception-Action Cycle, Radar and Radio*. Cambridge University Press, 2012.

[5] I. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, Apr 2008.

[6] M. Bkassiny, Y. Li, and S. Jayaweera, "A survey on machine-learning techniques in cognitive radios," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1136–1159, Third Quarter 2013.

[7] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys Tutorials*, vol. 11, no. 1, pp. 116–130, First Quarter 2009.

[8] "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical Communication*, vol. 4, no. 1, pp. 40 – 62, Mar. 2011.

[9] Y. Zeng, Y.-C. Liang, A. T. Hoang, and R. Zhang, "A review on spectrum sensing for cognitive radio: Challenges and solutions," *EURASIP J. Adv. Signal Process*, vol. 2010, pp. 2:2–2:2, Jan. 2010.

[10] P. Makris, D. Skoutas, and C. Skianis, "A survey on context-aware mobile and wireless networking: On networking and computing environments' integration," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 362–386, First Quarter 2013.

[11] P. Balamuralidhar and R. Prasad, "A context driven architecture for cognitive radio nodes," *Wireless Personal Communications*, vol. 45, no. 3, pp. 423–434, Mar. 2008.

[12] Z. Zhang, K. Long, and J. Wang, "Self-organization paradigms and optimization approaches for cognitive radio technologies: a survey," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 36–42, Apr. 2013.

[13] O. Aliu, A. Imran, M. Imran, and B. Evans, "A survey of self organisation in future cellular networks," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 336–361, First Quarter 2013.

[14] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Networks.*, vol. 9, no. 3, pp. 228–248, May 2011.

[15] S. Misra, M. Reisslein, and G. Xue, "A survey of multimedia streaming in wireless sensor networks," *IEEE Communications Surveys Tutorials*, vol. 10, no. 4, pp. 18–39, Fourth Quarter 2008.

[16] M. Chiang, S. Low, A. Calderbank, and J. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[17] E. Tragos, S. Zeadally, A. Fragkiadakis, and V. Siris, "Spectrum assignment in cognitive radio networks: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1108–1135, Third Quarter 2013.

[18] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjorungnes, *Game theory in wireless and communication networks*. Cambridge University Press, 2012.

[19] K. Akkarajitsakul, E. Hossain, D. Niyato, and D. I. Kim, "Game theoretic approaches for multiple access in wireless networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 13, no. 3, pp. 372–395, Third Quarter 2011.

[20] X. Liang and Y. Xiao, "Game theory for network security," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 472–486, First Quarter 2013.

[21] J. Wang, M. Peng, S. Jin, and C. Zhao, "A generalized nash equilibrium approach for robust cognitive radio networks via generalized variational inequalities," *IEEE Transactions on Wireless Communications*, vol. 13, no. 7, pp. 3701–3714, Jul. 2014.

[22] J. Ye, X. Shen, and J. W. Mark, "Call admission control in wideband cdma cellular networks by using fuzzy logic," *IEEE Transactions on Mobile Computing*, vol. 4, no. 2, pp. 129–141, March-April 2005.

[23] H. Mansour, P. Nasiopoulos, and V. Krishnamurthy, "Rate and distortion modeling of cgs coded scalable video content," *IEEE Transactions on Multimedia*, vol. 13, no. 2, pp. 165–180, Apr. 2011.

[24] W. Wang and A. Kwasinski, "Adaptive learning for scalable video transmission with harq over dynamic wireless channels," in *2015 IEEE International Conference on Communications*, London, UK, Jun. 2015, pp. 3094–3099.

[25] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, vol. 4, no. 1, pp. 237–285, May 1996.

[26] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 38, no. 2, pp. 156–172, Mar. 2008.

[27] M. Masonta, M. Mzyece, and N. Ntlatlapa, "Spectrum decision in cognitive radio networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1088–1107, Third Quarter 2013.

[28] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Computer Networks*, vol. 50, no. 13, pp. 2127 – 2159, Sep. 2006.

[29] H. A. A. Al-Rawi and K.-L. A. Yau, "Routing in distributed cognitive radio networks: A survey," *Wireless Personal Communications*, vol. 69, no. 4, pp. 1983–2020, Apr. 2013.

[30] X. Xu, C. Xiaomeng, and Z. Zhongshan, "Self-organization approaches for optimization in cognitive radio networks," *China Communications*, vol. 11, no. 4, pp. 121–129, Apr. 2014.

[31] K. R. Liu and B. Wang, *Cognitive radio networking and security: A game-theoretic view*. Cambridge University Press, 2010.

[32] A. He, K. K. Bae, T. Newman, J. Gaeddert, K. Kim, R. Menon, L. Morales-Tirado, J. Neel, Y. Zhao, J. Reed, and W. Tranter, "A survey of artificial intelligence for cognitive radios," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1578–1592, May 2010.

[33] D. Fudenberg, *The theory of learning in games*. MIT press, 1998.

[34] L. Buşoniu, R. Babuška, and B. Schutter, *Innovations in Multi-Agent Systems and Applications - 1*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, ch. Multi-agent Reinforcement Learning: An Overview, pp. 183–221.

[35] M. Thathachar and P. S. Sastry, "Varieties of learning automata: an overview," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 32, no. 6, pp. 711–722, Dec. 2002.

[36] S. Lasaulce and H. Tembine, *Game theory and learning for wireless networks: fundamentals and applications*. Academic Press, 2011.

[37] H. Tembine, *Distributed strategic learning for wireless engineers*. CRC Press, 2012.

[38] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal, "Reinforcement learning for context awareness and intelligence in wireless networks: Review, new features and open issues," *Journal of Network and Computer Applications*, vol. 35, no. 1, pp. 253 – 267, Jan. 2012.

[39] A. Forster, "Machine learning techniques applied to wireless ad-hoc networks: Guide and survey," in *3rd International Conference on Intelligent Sensors, Sensor Networks and Information, 2007*, Melbourne, Australia, Dec. 2007, pp. 365–370.

[40] M. Abu Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Communications Surveys Tutorials*, vol. 16, no. 4, pp. 1996–2018, Fourth Quarter 2014.

[41] M. Sato, K. Abe, and H. Takeda, "Learning control of finite markov chains with unknown transition probabilities," *IEEE Transactions on Automatic Control*, vol. 27, no. 2, pp. 502–505, Apr. 1982.

[42] C. J. C. H. Watkins, "Learning from delayed rewards," Ph.D. dissertation, King's College, Cambridge, UK, May 1989.

[43] O. Ibe, *Markov processes for stochastic modeling*. Academic press, 2008.

[44] C. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[45] S. Mahadevan, "Average reward reinforcement learning: Foundations, algorithms, and empirical results," *Machine Learning*, vol. 22, no. 1-3, pp. 159–195, Jan. 1996.

[46] K. S. Narendra and M. A. Thathachar, *Learning automata: an introduction*. Prentice-Hall, Inc., 1989.

[47] J. Wheeler, R. and K. Narendra, "Decentralized learning in finite markov chains," *IEEE Transactions on Automatic Control*, vol. 31, no. 6, pp. 519–526, Jun. 1986.

[48] T. I. Ahamed, P. N. Rao, and P. Sastry, "A reinforcement learning approach to automatic generation control," *Electric Power Systems Research*, vol. 63, no. 1, pp. 9 – 26, Aug. 2002.

[49] I. Grondman, L. Busoniu, G. Lopes, and R. Babuska, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 1291–1307, Nov. 2012.

[50] M. Wiering and H. van Hasselt, "Ensemble algorithms in reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 4, pp. 930–936, Aug. 2008.

[51] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 1998.

[52] M. Tan, "Multi-agent reinforcement learning: Independent vs. cooperative agents," in *Readings in Agents*, M. N. Huhns and M. P. Singh, Eds. San Francisco, CA: Morgan Kaufmann Publishers Inc., 1998, pp. 487–494.

[53] L. Panait and S. Luke, "Cooperative multi-agent learning: The state of the art," *Autonomous Agents and Multi-Agent Systems*, vol. 11, no. 3, pp. 387–434, Nov. 2005.

[54] M. Ahmadabadi and M. Asadpour, "Expertness based cooperative q-learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 32, no. 1, pp. 66–76, Feb. 2002.

[55] S. Sen and M. Sekaran, "Individual learning of coordination knowledge," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 10, no. 3, pp. 333–356, Jul. 1998.

[56] C. Claus and C. Boutilier, "The dynamics of reinforcement learning in cooperative multiagent systems," in *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*. Madison, WI: American Association for Artificial Intelligence, Jul. 1998, pp. 746–752.

[57] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems," *The Knowledge Engineering Review*, vol. 27, pp. 1–31, Mar. 2012.

[58] M. L. Littman, "Value-function reinforcement learning in markov games," *Cognitive Systems Research*, vol. 2, no. 1, pp. 55 – 66, Apr. 2002.

[59] M. Lauer and M. Riedmiller, "An algorithm for distributed reinforcement learning in cooperative multi-agent systems," in *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000.

[60] "Multiagent learning using a variable learning rate," *Artificial Intelligence*, vol. 136, no. 2, pp. 215 – 250, 2002.

[61] B. Price and C. Boutilier, "Accelerating reinforcement learning through implicit imitation," *Journal of Artificial Intelligence Research*, vol. 19, no. 1, pp. 569–629, Dec. 2003.

[62] A. B. MacKenzie and L. A. DaSilva, *Game Theory for Wireless Engineers (Synthesis Lectures on Communications)*. Morgan & Claypool Publishers, 2006.

[63] N. Vieille, "Chapter 48 stochastic games: Recent results," ser. Handbook of Game Theory with Economic Applications, R. Aumann and S. Hart, Eds. Elsevier, 2002, vol. 3, pp. 1833 – 1850.

[64] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proceedings of the 11th International Conference on Machine Learning*. New Brunswick, NJ: Morgan Kaufmann, Jul. 1994, pp. 157–163.

[65] M. Weinberg and J. S. Rosenschein, "Best-response multiagent learning in non-stationary environments," in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*. New York, NY: IEEE Computer Society, Jul. 2004, pp. 506–513.

[66] J. Hu and M. P. Wellman, "Nash q-learning for general-sum stochastic games," *Journal of Machine Learning Research*, vol. 4, pp. 1039–1069, Dec. 2003.

[67] J. Li, K. Ramachandran, and T. K. Das, "A reinforcement learning (nash-r) algorithm for average reward irreducible stochastic games," *Journal of Machine Learning Research*, 2007.

[68] A. Greenwald, M. Zinkevich, and P. Kaelbling, "Correlated q-learning," in *Proceedings of the Twentieth International Conference on Machine Learning*, Aug. 2003, pp. 242–249.

[69] D. M. Topkis, "Equilibrium points in nonzero-sum n-person submodular games," *SIAM Journal on Control and Optimization*, vol. 17, no. 6, pp. 773–787, Dec. 1979.

[70] A. Poznyak and K. Najim, "Learning through reinforcement for n-person repeated constrained games," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 32, no. 6, pp. 759–771, Dec. 2002.

[71] J. Shamma and G. Arslan, "Dynamic fictitious play, dynamic gradient play, and distributed convergence to nash equilibria," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 312–327, Mar. 2005.

[72] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, Sep. 2000.

[73] D. S. Leslie and E. J. Collins, "Convergent multiple-timescales reinforcement learning algorithms in normal form games," *The Annals of Applied Probability*, vol. 13, no. 4, pp. pp. 1231–1251, Nov. 2003.

[74] A. Jean-Marie and M. Tidball, "Adapting behaviors through a learning process," *Journal of Economic Behavior & Organization*, vol. 60, no. 3, pp. 399 – 422, Jul. 2006.

[75] M. Wellman and J. Hu, "Conjectural equilibrium in multiagent learning," *Machine Learning*, vol. 33, no. 2-3, pp. 179–200, Nov. 1998.

[76] J. Hofbauer and W. H. Sandholm, "On the global convergence of stochastic fictitious play," *Econometrica*, vol. 70, no. 6, pp. 2265–2294, Nov. 2002.

[77] J. Marden, G. Arslan, and J. Shamma, "Cooperative control and potential games," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 6, pp. 1393–1407, Dec. 2009.

[78] A. Jafari, A. R. Greenwald, D. Gondek, and G. Ercal, "On no-regret learning, fictitious play, and nash equilibrium," in *Proceedings of the Eighteenth International Conference on Machine Learning*. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2001, pp. 226–233.

[79] P. Vrancx, K. Verbeeck, and A. Nowe, "Decentralized learning in markov games," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 4, pp. 976–981, Aug. 2008.

[80] J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Advances in Neural Information Processing Systems*, vol. 6. Morgan Kaufmann, 1994, pp. 671–678.

[81] J. Nie and S. Haykin, "A q-learning-based dynamic channel assignment technique for mobile communication systems," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1676–1687, Sep. 1999.

[82] ——, "A dynamic channel assignment policy through q-learning," *IEEE Transactions on Neural Networks*, vol. 10, no. 6, pp. 1443–1455, Nov. 1999.

[83] Y.-S. Chen, C.-J. Chang, and F.-C. Ren, "Q-learning-based multirate transmission control scheme for rrm in multimedia wcdma systems," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 1, pp. 38–48, Jan. 2004.

[84] L. Li, T. J. Walsh, and M. L. Littman, "Towards a unified theory of state abstraction for mdps," in *Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics*, 2006, pp. 531–539.

[85] N. Lilith and K. Dogancay, "Dynamic channel allocation for mobile cellular traffic using reduced-state reinforcement learning," in *2004 IEEE Wireless Communications and Networking Conference, 2004*, vol. 4, March 2004, pp. 2195–2200 Vol.4.

[86] ——, "Distributed reduced-state sarsa algorithm for dynamic channel allocation in cellular networks featuring traffic mobility," in *IEEE International Conference on Communications*, vol. 2, Seoul, Korea, May 2005.

[87] M. van der Schaar and F. Fu, "Spectrum access games and strategic learning in cognitive radio networks for delay-critical applications," *Proceedings of the IEEE*, vol. 97, no. 4, pp. 720–740, Apr. 2009.

[88] F. Fu and M. van der Schaar, "Learning to compete for resources in wireless stochastic games," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1904–1919, May 2009.

[89] Y. Zhang, F. Fu, and M. van der Schaar, "On-line learning and optimization for wireless video transmission," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3108–3124, Jun. 2010.

[90] Q. Zhao, S. Geirhofer, L. Tong, and B. Sadler, "Opportunistic spectrum access via periodic channel sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 785–796, Feb. 2008.

[91] A. Karmokar, D. Djonin, and V. Bhargava, "Cross-layer rate and power adaptation strategies for ir-harq systems over fading channels with memory: A smdp-based approach," *IEEE Transactions on Communications*, vol. 56, no. 8, pp. 1352–1365, Aug. 2008.

[92] P. de Cuetos and K. Ross, "Unified framework for optimal video streaming," in *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, Hong Kong, China, Mar. 2004, pp. 1479–1489 vol.3.

[93] N. Mastronarde and M. van der Schaar, "Online reinforcement learning for dynamic multimedia systems," *IEEE Transactions on Image Processing*, vol. 19, no. 2, pp. 290–305, Feb. 2010.

[94] Y. Fei, V. W. S. Wong, and V. C. M. Leung, "Efficient qos provisioning for adaptive multimedia in mobile communication networks by reinforcement learning," *Journal of Mobile Networks and Applications*, vol. 11, no. 1, pp. 101–110, Feb. 2006.

[95] M. Levorato, S. Firouzabadi, and A. Goldsmith, "A reinforcement learning optimization framework for cognitive interference networks," in *49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Urbana, IL, Sep. 2011, pp. 1633–1640.

[96] K. Phan, T. Le-Ngoc, M. van der Schaar, and F. Fu, "Optimal scheduling over time-varying channels with traffic admission control: Structural results and online learning algorithms," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4434–4444, Sep. 2013.

[97] S. Firouzabadi, M. Levorato, D. O'Neill, and A. Goldsmith, "Learning interference strategies in cognitive arq networks," in *2010 IEEE Global Telecommunications Conference*, Miami, FL, Dec. 2010.

[98] C. Sun, E. Stevens-Navarro, V. Shah-Mansouri, and V. W. Wong, "A constrained mdp-based vertical handoff decision algorithm for 4g heterogeneous wireless networks," *Wireless Networks*, vol. 17, no. 4, pp. 1063–1081, May 2011.

[99] F. Yu, V. Wong, and V. Leung, "A new qos provisioning method for adaptive multimedia in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1899–1909, May 2008.

[100] E. Altman, *Constrained Markov decision processes*. CRC Press, 1999, vol. 7.

[101] A. Pietrabissa, "A reinforcement learning approach to call admission and call dropping control in links with variable capacity," *European Journal of Control*, vol. 17, no. 1, pp. 89–103, Jan. 2011.

[102] D. Djonin and V. Krishnamurthy, "Q-learning algorithms for constrained markov decision processes with randomized monotone policies: Application to mimo transmission control," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 2170–2181, May 2007.

[103] E.-S. El-Alfy, Y.-D. Yao, and H. Heffes, "A learning approach for prioritized handoff channel allocation in mobile multimedia networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 7, pp. 1651–1660, Jul. 2006.

[104] F. Yu, V. Wong, and V. Leung, "A new qos provisioning method for adaptive multimedia in cellular wireless networks," in *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, Hong Kong, China, Mar. 2004, pp. 2130–2141 vol.3.

[105] W. Usaha and J. Barria, "Qos routing in manets with imprecise information using actor-critic reinforcement learning," in *IEEE Wireless Communications and Networking Conference*, Mar. 2007, pp. 3382–3387.

[106] U. Berthold, F. Fu, M. van der Schaar, and F. Jondral, "Detection of spectral resources in cognitive radios using reinforcement learning," in *3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Chicago, IL, Oct. 2008.

[107] M. Levorato, S. Firouzabadi, and A. Goldsmith, "A learning framework for cognitive interference networks with partial and noisy observations," *IEEE Transactions on Wireless Communications*, vol. 11, no. 9, pp. 3101–3111, Sep. 2012.

[108] Y.-H. Chen, C.-J. Chang, and C. Y. Huang, "Fuzzy q-learning admission control for wcdma/wlan heterogeneous networks with multimedia traffic," *IEEE Transactions on Mobile Computing*, vol. 8, no. 11, pp. 1469–1479, Nov. 2009.

[109] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive mac for opportunistic spectrum access in ad hoc networks: A pomdp framework," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 3, pp. 589–600, Apr. 2007.

[110] V. R. Konda and V. S. Borkar, "Actor-critic–type learning algorithms for markov decision processes," *SIAM Journal on Control and Optimization*, vol. 38, no. 1, pp. 94–123, Nov. 1999.

[111] J. Baxter and P. L. Bartlett, "Reinforcement learning in pomdp's via direct gradient ascent," in *Proceedings of the Seventeenth International Conference on Machine Learning*. San Francisco, CA: Morgan Kaufmann Publishers Inc., Jun. 2000, pp. 41–48.

[112] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, "A survey on 3gpp heterogeneous networks," *IEEE Wireless Communications*, vol. 18, no. 3, pp. 10–21, Jun. 2011.

[113] A. Galindo-Serrano and L. Giupponi, "Distributed q-learning for interference control in ofdma-based femtocell networks," in *2010 IEEE 71st Vehicular Technology Conference*, Taipei, Taiwan, May 2010.

[114] M. Simsek, M. Bennis, and A. Czylwik, "Dynamic inter-cell interference coordination in hetnets: A reinforcement learning approach," in *2012 IEEE Global Communications Conference*, Anaheim, CA, Dec. 2012, pp. 5446–5450.

[115] M. Bennis, S. Perlaza, P. Blasco, Z. Han, and H. Poor, "Self-organization in small cell networks: A reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 12, no. 7, pp. 3202–3212, Jul. 2013.

[116] W. Wang, A. Kwasinski, and Z. Han, "Power allocation with stackelberg game in femtocell networks: A self-learning approach," in *2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking*, Singapore, Jun. 2014, pp. 354–362.

[117] A. Galindo-Serrano and L. Giupponi, "Distributed q-learning for aggregated interference control in cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1823–1834, May 2010.

[118] M. Bennis and D. Niyato, "A q-learning based approach to interference avoidance in self-organized femtocell networks," in *IEEE GLOBECOM Workshops*, Miami, FL, Dec. 2010, pp. 706–710.

[119] C. Wu, K. Chowdhury, M. Di Felice, and W. Meleis, "Spectrum management of cognitive radio using multi-agent reinforcement learning," in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Industry Track*, ser. AAMAS '10. Toronto, Canada: International Foundation for Autonomous Agents and Multiagent Systems, 2010, pp. 1705–1712.

[120] K.-L. Yau, P. Komisarczuk, and P. Teal, "Context-awareness and intelligence in distributed cognitive radio networks: A reinforcement learning

approach," in *2010 Australian Communications Theory Workshop*, Canberra, ACT, Australia, Feb. 2010.

[121] O. van den Biggelaar, J.-M. Dricot, P. De Doncker, and F. Horlin, "Sensing time and power allocation for cognitive radios using distributed q-learning," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, Apr. 2012.

[122] X. Chen, Z. Zhao, H. Zhang, and T. Chen, "Applying multi-agent q-learning scheme in cognitive wireless mesh networks for green communications," in *2010 IEEE 21st International Symposium on Personal, Indoor and Mobile Radio Communications Workshops*, Instanbul, Turkey, Sep. 2010.

[123] N. Morozs, T. Clarke, D. Grace, and Q. Zhao, "Distributed q-learning based dynamic spectrum management in cognitive cellular systems: Choosing the right learning rate," in *2014 IEEE Symposium on Computers and Communication*, Funchal, Portugal, Jun. 2014.

[124] H. Liu, B. Krishnamachari, and Q. Zhao, "Cooperation and learning in multiuser opportunistic spectrum access," in *IEEE International Conference on Communications Workshops*, Beijing, May 2008, pp. 487–492.

[125] M. NoroozOliaee, B. Hamdaoui, and K. Tumer, "Efficient objective functions for coordinated learning in large-scale distributed osa systems," *IEEE Transactions on Mobile Computing*, vol. 12, no. 5, pp. 931–944, May 2013.

[126] H. Li, "Multiagent q-learning for aloha-like spectrum access in cognitive radio systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, pp. 56:1–56:13, Apr. 2010.

[127] H. Robbins and S. Monro, "A stochastic approximation method," *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400–407, Sep. 1951.

[128] M. Seah, C.-K. Tham, V. Srinivasan, and A. Xin, "Achieving coverage through distributed reinforcement learning in wireless sensor networks," in *3rd International Conference on Intelligent Sensors, Sensor Networks and Information*, Melbourne, Australia, Dec. 2007, pp. 425–430.

[129] G. Naddafzadeh-Shirazi, P.-Y. Kong, and C.-K. Tham, "Distributed reinforcement learning frameworks for cooperative retransmission in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 8, pp. 4157–4162, Oct. 2010.

[130] W. Wang and A. Kwasinski, "Experience cooperative sharing in cross-layer cognitive radio for real-time multimedia communication," in *the 4th International Conference on Cognitive Radio and Advanced Spectrum Management*, ser. CogART '11, Barcelona, Spain, 2011, pp. 55:1–55:5.

[131] L. Giupponi, A. Galindo-Serrano, P. Blasco, and M. Dohler, "Docitive networks: an emerging paradigm for dynamic spectrum management," *IEEE Wireless Communications*, vol. 17, no. 4, pp. 47–54, Aug. 2010.

[132] A. Galindo-Serrano, L. Giupponi, P. Blasco, and M. Dohler, "Learning from experts in cognitive radio networks: The docitive paradigm," in *Proceedings of the Fifth International Conference on Cognitive Radio Oriented Wireless Networks Communications*, Cannes, France, Jun. 2010.

[133] L. Giupponi, A. M. Galindo-Serrano, and M. Dohler, "From cognition to docition: The teaching radio paradigm for distributed & autonomous deployments," *Computer Communications*, vol. 33, no. 17, pp. 2015–2020, Nov. 2010.

[134] A. Galindo-Serrano, L. Giupponi, and M. Dohler, "Cognition and docition in ofdma-based femtocell networks," in *2010 IEEE Global Telecommunications Conference*, Miami, FL, Dec 2010.

[135] J. Tefft and N. Kirsch, "Accelerated learning in machine learning-based resource allocation methods for heterogenous networks," in *IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems*, Cannes, France, Sep. 2013.

[136] C. Saraydar, N. B. Mandayam, and D. Goodman, "Efficient power control via pricing in wireless data networks," *IEEE Transactions on Communications*, vol. 50, no. 2, pp. 291–303, Feb. 2002.

[137] A. Imran, M. Bennis, and L. Giupponi, "Use of learning, game theory and optimization as biomimetic approaches for self-organization in macro-femtocell coexistence," in *2012 IEEE Wireless Communications and Networking Conference Workshops*, Paris, France, Apr. 2012.

[138] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *Journal of Machine Learning Research*, vol. 10, pp. 1633–1685, Dec. 2009.

[139] C. Camerer and T. Hua Ho, "Experience-weighted attraction learning in normal form games," *Econometrica*, vol. 67, no. 4, pp. 827–874, Jul. 1999.

[140] H.-P. Shiang and M. van der Schaar, "Distributed resource management in multihop cognitive radio networks for delay-sensitive transmission,"

[141] Q. Zhu, W. Saad, Z. Han, H. Poor, and T. Basar, "Eavesdropping and jamming in next-generation wireless networks: A game-theoretic approach," in *IEEE Military Communication Conference*, Baltimore, MD, Nov. 2011, pp. 119–124.

[142] U. Candogan, I. Menache, A. Ozdaglar, and P. Parrilo, "Near-optimal power control in wireless networks: A potential game approach," in *Twenty-ninth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Diego, CA, Mar. 2010.

[143] W. Saad, Q. Zhu, T. Basar, Z. Han, and A. Hjorungnes, "Hierarchical network formation games in the uplink of multi-hop wireless networks," in *IEEE Global Telecommunications Conference*, Honolulu, HI, Nov. 2009.

[144] M. Khan, H. Tembine, and A. Vasilakos, "Game dynamics and cost of learning in heterogeneous 4g networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 1, pp. 198–213, Jan. 2012.

[145] C. Long, Q. Zhang, B. Li, H. Yang, and X. Guan, "Non-cooperative power control for wireless ad hoc networks with repeated games," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 6, pp. 1101–1112, Aug. 2007.

[146] G. Arslan and J. Shamma, "Distributed convergence to nash equilibria with local utility measurements," in *43rd IEEE Conference on Decision and Control*, vol. 2, Lost Angeles, CA, Dec. 2004, pp. 1538–1543 Vol.2.

[147] T. Cui, L. Chen, and S. Low, "A game-theoretic framework for medium access control," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 7, pp. 1116–1127, Sep. 2008.

[148] T. Alpcan and T. Basar, "A hybrid systems model for power control in multicell wireless data networks," *Performance Evaluation*, vol. 57, no. 4, pp. 477–495, Aug. 2004.

[149] G. Arslan, M. Demirkol, and Y. Song, "Equilibrium efficiency improvement in mimo interference systems: A decentralized stream control approach," *IEEE Transactions on Wireless Communications*, vol. 6, no. 8, pp. 2984–2993, Aug. 2007.

[150] A. Ali, J. Qadir, and A. Baig, "Learning automata based multipath multicasting in cognitive radio networks," *Journal of Communications and Networks*, vol. 17, no. 4, pp. 406–418, Aug. 2015.

[151] J. A. Torkestani and M. R. Meybodi, "Mobility-based multicast routing algorithm for wireless mobile ad-hoc networks: A learning automata approach," *Computer Communications*, vol. 33, no. 6, pp. 721 – 735, Apr. 2010.

[152] P. S. Sastry, V. V. Phansalkar, and M. Thathachar, "Decentralized learning of nash equilibria in multi-person stochastic games with incomplete information," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 5, pp. 769–777, May 1994.

[153] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in unknown dynamic environment: A game-theoretic stochastic learning solution," *IEEE Transactions on Wireless Communications,*, vol. 11, no. 4, pp. 1380–1391, Apr. 2012.

[154] W. Zhong, G. Chen, S. Jin, and K.-K. Wong, "Relay selection and discrete power control for cognitive relay networks via potential game," *IEEE Transactions on Signal Processing*, vol. 62, no. 20, pp. 5411–5424, Oct. 2014.

[155] J. Zheng, Y. Cai, N. Lu, Y. Xu, and X. Shen, "Stochastic game-theoretic spectrum access in distributed and dynamic environment," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4807–4820, Oct. 2015.

[156] W. Zhong, Y. Xu, and M. Tao, "Precoding strategy selection for cognitive mimo multiple access channels using learning automata," in *IEEE International Conference on Communications*, Cape Town, South Africa, May 2010.

[157] Y. Xing and R. Chandramouli, "Stochastic learning solution for distributed discrete power control game in wireless data networks," *IEEE/ ACM Transactions on Networking*, vol. 16, no. 4, pp. 932–944, Aug. 2008.

[158] Y. Song, C. Zhang, and Y. Fang, "Stochastic traffic engineering in multihop cognitive wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 3, pp. 305–316, Mar. 2010.

[159] P. Zhou, Y. Chang, and J. Copeland, "Reinforcement learning for repeated power control game in cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 1, pp. 54–69, Jan. 2012.

[160] A. Poznyak and K. Najim, *Learning Automata and Stochastic Optimization*. Springer, 1997.

[161] A. Greenwald and A. Jafari, "A general class of no-regret learning algorithms and game-theoretic equilibria," in *Learning Theory and*

[140] (cont.) *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 941–953, Feb. 2009.

Kernel Machines, ser. Lecture Notes in Computer Science, B. Schlkopf and M. Warmuth, Eds. Springer Berlin Heidelberg, 2003, vol. 2777, pp. 2–12.

[162] N. Nie and C. Comaniciu, "Adaptive channel allocation spectrum etiquette for cognitive radio networks," *Mobile Networks and Applications*, vol. 11, no. 6, pp. 779–797, Dec. 2006.

[163] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic game theory*. Cambridge University Press, 2007.

[164] Q. Zhu, Z. Han, and T. Basar, "No-regret learning in collaborative spectrum sensing with malicious nodes," in *IEEE International Conference on Communications*, Cape Town, South Africa, May 2010.

[165] Z. Han, C. Pandana, and K. Liu, "Distributive opportunistic spectrum access for cognitive radio using correlated equilibrium and no-regret learning," in *IEEE Wireless Communications and Networking Conference*, Hong Kong, China, Mar. 2007, pp. 11–15.

[166] L. Chen, S. Iellamo, M. Coupechoux, and P. Godlewski, "An auction framework for spectrum allocation with interference constraint in cognitive radio networks," in *Twenty-ninth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Diego, CA, Mar. 2010.

[167] S. Maharjan, Y. Zhang, C. Yuen, and S. Gjessing, "Distributed spectrum sensing in cognitive radio networks with fairness consideration: Efficiency of correlated equilibrium," in *IEEE 8th International Conference on Mobile Adhoc and Sensor Systems*, Valencia, Spain, Oct. 2011, pp. 540–549.

[168] J. Zheng, Y. Cai, and D. Wu, "Subcarrier allocation based on correlated equilibrium in multi-cell ofdma systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 1–12, Jul. 2012.

[169] R. Nau, S. G. Canovas, and P. Hansen, "On the geometry of nash equilibria and correlated equilibria," *International Journal of Game Theory*, vol. 32, no. 4, pp. 443–453, Aug. 2004.

[170] M. Maskery, V. Krishnamurthy, and Q. Zhao, "Decentralized dynamic spectrum access for cognitive radios: cooperative design of a non-cooperative game," *IEEE Transactions on Communications*, vol. 57, no. 2, pp. 459–469, Feb. 2009.

[171] J. Zheng, Y. Cai, Y. Xu, and A. Anpalagan, "Distributed channel selection for interference mitigation in dynamic environment: A game-theoretic stochastic learning solution," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4757–4762, Nov. 2014.

[172] M. Bowling and M. Veloso, "An analysis of stochastic game theory for multiagent reinforcement learning," Computer Science Department, Carnegie Mellon University, Tech. Rep., 2000.

[173] E. Altman, K. Avratchenkov, N. Bonneau, M. Debbah, R. El-Azouzi, and D. Menasche, "Constrained stochastic games in wireless networks," in *IEEE Global Telecommunications Conference*, Washington, DC, Nov. 2007, pp. 315–320.

[174] B. Wang, Y. Wu, K. Liu, and T. Clancy, "An anti-jamming stochastic game for cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 877–889, Apr. 2011.

[175] Y. Gwon, S. Dastangoo, C. Fossa, and H. Kung, "Competing mobile network game: Embracing antijamming and jamming strategies with reinforcement learning," in *IEEE Conference on Communications and Network Security*, National Harbor, MD, Oct. 2013, pp. 28–36.

[176] C. Chen, M. Song, C. Xin, and J. Backens, "A game-theoretical anti-jamming scheme for cognitive radio networks," *IEEE Network*, vol. 27, no. 3, pp. 22–27, May 2013.

[177] S. Sarkar and R. Datta, "A game theoretic model for stochastic routing in self-organized manets," in *IEEE Wireless Communications and Networking Conference*, Shanghai, China, Apr. 2013, pp. 1962–1967.

[178] F. Fu and U. C. Kozat, "Stochastic game for wireless network virtualization," *IEEE/ACM Transactions on Networking*, vol. 21, no. 1, pp. 84–97, Feb. 2013.

[179] X. Chen, J. Wu, Y. Cai, H. Zhang, and T. Chen, "Energy-efficiency oriented traffic offloading in wireless networks: A brief survey and a learning approach for heterogeneous cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 4, pp. 627–640, Apr. 2015.

[180] J. W. Huang and V. Krishnamurthy, "Game theoretic issues in cognitive radio systems (invited paper)," *Journal of Communications*, vol. 4, no. 10, Nov. 2009.

[181] W. Wang and A. Kwasinski, "Distributed cross-layer resource allocation using correlated equilibrium based stochastic learning," in *IEEE Wireless Communications and Networking Conference*, Shanghai, China, Apr. 2013.

[182] Y. Su and M. van der Schaar, "Dynamic conjectures in random access networks using bio-inspired learning," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 587–601, May 2010.

[183] X. Chen, Z. Zhao, and H. Zhang, "Stochastic power adaptation with multiagent reinforcement learning for cognitive wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2155–2166, Nov. 2013.

[184] Y. Cao, D. Duan, X. Cheng, L. Yang, and J. Wei, "Qos-oriented wireless routing for smart meter data collection: Stochastic learning on graph," *IEEE Transactions on Wireless Communications*, vol. 13, no. 8, pp. 4470–4482, Aug. 2014.

[185] S. Ganzfried and T. Sandholm, "Computing equilibria in multiplayer stochastic games of imperfect information," in *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, ser. IJCAI'09. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2009, pp. 140–146.

[186] X. Liu, G. Ding, Y. Yang, Q. Wu, and J. Wang, "A stochastic game framework for joint frequency and power allocation in dynamic decentralized cognitive radio networks," *AEU - International Journal of Electronics and Communications*, vol. 67, no. 10, pp. 817 – 826, Oct. 2013.

[187] Q. Zhu, Z. Yuan, J. B. Song, Z. Han, and T. Basar, "Interference aware routing game for cognitive radio multi-hop networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 10, pp. 2006–2015, Nov. 2012.

[188] J. Huang and V. Krishnamurthy, "Transmission control in cognitive radio as a markovian dynamic game: Structural result on randomized threshold policies," *IEEE Transactions on Communications*, vol. 58, no. 1, pp. 301–310, Jan. 2010.

[189] J. Huang, H. Mansour, and V. Krishnamurthy, "A dynamical games approach to transmission-rate adaptation in multimedia wlan," *IEEE Transactions on Signal Processing*, vol. 58, no. 7, pp. 3635–3646, Jul. 2010.

[190] J. C. Spall, *Introduction to stochastic search and optimization: estimation, simulation, and control*. John Wiley & Sons, 2005, vol. 65.

[191] Q. Zhu, H. Tembine, and T. Baar, *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. John Wiley & Sons, Inc., 2013, ch. Hybrid Learning in Stochastic Games and Its Application in Network Security, pp. 303–329.

[192] A. Hanif, H. Tembine, M. Assaad, and D. Zeghlache, "On the convergence of a nash seeking algorithm with stochastic state dependent payoff," *arXiv preprint arXiv:1210.0193*, 2012.

[193] Q. Zhu, H. Tembine, and T. Basar, "Distributed strategic learning with application to network security," in *American Control Conference (ACC), 2011*, Jun. 2011, pp. 4057–4062.

[194] Y. Shoham, R. Powers, and T. Grenager, "If multi-agent learning is the answer, what is the question?" *Artificial Intelligence*, vol. 171, no. 7, pp. 365 – 377, May 2007, foundations of Multi-Agent Learning.

[195] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in cognitive radio networks: Global optimization using local interaction games," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 2, pp. 180–194, Apr. 2012.

[196] Q. D. La, Y. Chew, and B. H. Soong, "An interference-minimization potential game for ofdma-based distributed spectrum sharing systems," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3374–3385, Sep. 2011.

[197] X. Chen, H. Zhang, T. Chen, and M. Lasanen, "Improving energy efficiency in green femtocell networks: A hierarchical reinforcement learning framework," in *2013 IEEE International Conference on Communications*, Budapest, Hungary, Jun. 2013.

[198] S. Zhong and Y. Zhang, "How to select optimal gateway in multi-domain wireless networks: Alternative solutions without learning," *IEEE Transactions on Wireless Communications*, vol. 12, no. 11, pp. 5620–5630, Nov. 2013.

[199] L. Rose, S. Perlaza, C. Le Martret, and M. Debbah, "Self-organization in decentralized networks: A trial and error learning approach," *IEEE Transactions on Wireless Communications*, vol. 13, no. 1, pp. 268–279, Jan. 2014.

[200] C. F. Camerer, T.-H. Ho, and J.-K. Chong, "Sophisticated experience-weighted attraction learning and strategic teaching in repeated games," *Journal of Economic Theory*, vol. 104, no. 1, pp. 137–188, May 2002.

[201] Q. Zhao, T. Jiang, N. Morozs, D. Grace, and T. Clarke, "Transfer learning: A paradigm for dynamic spectrum and topology management in flexible architectures," in *IEEE 78th Vehicular Technology Conference*, Las Vegas, NV, Sep. 2013.

[202] 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 Transactions on Wireless Communications*, vol. 13, no. 4, pp. 2000–2011, Apr. 2014.

[203] D. Cooper and J. Kagel, "Learning and transfer in signaling games," *Economic Theory*, vol. 34, no. 3, pp. 415–439, Mar. 2008.

[204] M. E. Taylor, *Transfer in Reinforcement Learning Domains*. Springer, 2009.

[205] H. Lateef, A. Imran, M. Ali Imran, L. Giupponi, and M. Dohler, "Lte-advanced self-organizing network conflicts and coordination algorithms," *IEEE Wireless Communications*, vol. 22, no. 3, pp. 108–117, Jun. 2015.

[206] H. Y. Lateef, A. Imran, and A. Abu-Dayya, "A framework for classification of self-organising network conflicts and coordination algorithms," in *2013 IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications*, Sep. 2013, pp. 2898–2903.

[207] A. Barto and S. Mahadevan, "Recent advances in hierarchical reinforcement learning," *Discrete Event Dynamic Systems*, vol. 13, no. 1-2, pp. 41–77, Jan. 2003.

[208] M. Ghavamzadeh, S. Mahadevan, and R. Makar, "Hierarchical multi-agent reinforcement learning," *Autonomous Agents and Multi-Agent Systems*, vol. 13, no. 2, pp. 197–229, Apr. 2006.