A Survey on Machine-Learning Techniques in Cognitive Radios

Mario Bkassiny, *Student Member, IEEE*, Yang Li, *Student Member, IEEE*, and Sudharman K. Jayaweera, *Senior Member, IEEE*

***Abstract*—In this survey paper, we characterize the learning problem in cognitive radios (CRs) and state the importance of artificial intelligence in achieving real cognitive communications systems. We review various learning problems that have been studied in the context of CRs classifying them under two main categories: Decision-making and feature classification. Decision-making is responsible for determining policies and decision rules for CRs while feature classification permits identifying and classifying different observation models. The learning algorithms encountered are categorized as either supervised or unsupervised algorithms. We describe in detail several challenging learning issues that arise in cognitive radio networks (CRNs), in particular in non-Markovian environments and decentralized networks, and present possible solution methods to address them. We discuss similarities and differences among the presented algorithms and identify the conditions under which each of the techniques may be applied.**

***Index Terms*—Artificial intelligence, cognitive radio, decision-making, feature classification, machine learning, supervised learning, unsupervised learning, .**

I. Introduction

THE TERM cognitive radio (CR) has been used to refer to radio devices that are capable of learning and adapting to their environment [1], [2]. Cognition, from the Latin word *cognoscere* (to know), is defined as a process involved in gaining knowledge and comprehension, including thinking, knowing, remembering, judging and problem solving [3]. A key aspect of any CR is the ability for self-programming or autonomous learning [4], [5]. In [6], Haykin envisioned CRs to be *brain-empowered* wireless devices that are specifically aimed at improving the utilization of the electromagnetic spectrum. According to Haykin, a CR is assumed to use the methodology of *understanding-by-building* and is aimed to achieve two primary objectives: Permanent reliable communications and efficient utilization of the spectrum resources [6]. With this interpretation of CRs, a new era of CRs began, focusing on dynamic spectrum sharing (DSS) techniques to improve the utilization of the crowded RF spectrum [6]–[10]. This led to research on various aspects of communications and signal processing required for dynamic spectrum access (DSA) networks [6], [11]–[38]. These included underlay, overlay and interweave paradigms for spectrum co-existence by secondary CRs in licensed spectrum bands [10].

To perform its cognitive tasks, a CR should be aware of its RF environment. It should sense its surrounding environment and identify all types of RF activities. Thus, spectrum sensing was identified as a major ingredient in CRs [6]. Many sensing techniques have been proposed over the last decade [15], [39], [40], based on matched filter, energy detection, cyclostationary detection, wavelet detection and covariance detection [30], [41]–[46]. In addition, cooperative spectrum sensing was proposed as a means of improving the sensing accuracy by addressing the hidden terminal problems inherent in wireless networks in [15], [33], [34], [42], [47]–[49]. In recent years, cooperative CRs have also been considered in literature as in [50]–[53]. Recent surveys on CRs can be found in [41], [54], [55]. A survey on the spectrum sensing techniques for CRs can be found in [39]. Several surveys on the DSA techniques and the medium access control (MAC) layer operations for the CRs are provided in [56]–[60].

In addition to being aware of its environment, and in order to be really *cognitive*, a CR should be equipped with the abilities of learning and reasoning [1], [2], [5], [61], [62]. These capabilities are to be embedded in a cognitive engine which has been identified as the core of a CR [63]–[68], following the pioneering vision of [2]. The cognitive engine is to coordinate the actions of the CR by making use of machine learning algorithms. However, only in recent years there has been a growing interest in applying machine learning algorithms to CRs [38], [69]–[72].

In general, learning becomes necessary if the precise effects of the inputs on the outputs of a given system are not known [69]. In other words, if the input-output function of the system is unknown, learning techniques are required to estimate that function in order to design proper inputs. For example, in wireless communications, the wireless channels are non-ideal and may cause uncertainty. If it is desired to reduce the probability of error over a wireless link by reducing the coding rate, learning techniques can be applied to estimate the wireless channel characteristics and to determine the specific coding rate that is required to achieve a certain probability of error [69]. The problem of channel estimation is relatively simple and can be solved via estimation algorithms [73]. However, in the case of CRs and cognitive radio networks (CRNs), problems become more complicated with the increase in the degrees of freedom of wireless systems especially with the introduction of highly-reconfigurable software-defined radios

Manuscript received 27 January 2012; revised 7 July 2012. This work was supported in part by the National Science foundation (NSF) under the grant CCF-0830545.

M. Bkassiny, Y. Li and S. K. Jayaweera are with the Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM, USA (e-mail: {bkassiny, yangli, jayaweera}@ece.unm.edu).

Digital Object Identifier 10.1109/SURV.2012.100412.00017

(SDRs). In this case, several parameters and policies need to be adjusted simultaneously (e.g. transmit power, coding scheme, modulation scheme, sensing algorithm, communication protocol, sensing policy, etc.) and no simple formula may be able to determine these setup parameters simultaneously. This is due to the complex interactions among these factors and their impact on the RF environment. Thus, learning methods can be applied to allow efficient adaption of the CRs to their environment, yet without the complete knowledge of the dependence among these parameters [74]. For example, in [71], [75], threshold-learning algorithms were proposed to allow CRs to reconfigure their spectrum sensing processes under uncertainty conditions.

The problem becomes even more complicated with heterogeneous CRNs. In this case, a CR not only has to adapt to the RF environment, but also it has to coordinate its actions with respect to the other radios in the network. With only a limited amount of information exchange among nodes, a CR needs to estimate the behavior of other nodes in order to select its proper actions. For example, in the context of DSA, CRs try to access idle primary channels while limiting collisions with both licensed and other secondary cognitive users [38]. In addition, if the CRs are operating in unknown RF environments [5], conventional solutions to the decision process (i.e. Dynamic Programming in the case of Markov Decision Processes (MDPs) [76]) may not be feasible since they require complete knowledge of the system. On the other hand, by applying special learning algorithms such as the reinforcement learning (RL) [38], [74], [77], it is possible to arrive at the optimal solution to the MDP, without knowing the transition probabilities of the Markov model. Therefore, given the reconfigurability requirements and the need for autonomous operation in unknown and heterogeneous RF environment, CRs may use learning algorithms as a tool for adaptation to the environment and to coordinate with peer radio devices. Moreover, incorporation of low-complexity learning algorithms can lead to reduced system complexities in CRs.

A look at the recent literature on CRs reveals that both supervised and unsupervised learning techniques have been proposed for various learning tasks. The authors in [65], [78], [79] have considered supervised learning based on neural networks and support vector machines (SVMs) for CR applications. On the other hand, unsupervised learning, such as RL, has been considered in [80], [81] for DSS applications. The distributed Q-learning algorithm has been shown to be effective in a particular CR application in [77]. For example, in [82], CRs used the Q-learning to improve detection and classification performance of primary signals. Other applications of RL to CRs can be found, for example, in [14], [83]–[85]. Recent work in [86] introduces novel approaches to improve the efficiency of RL by adopting a weight-driven exploration. Unsupervised Bayesian non-parametric learning based on the Dirichlet process was proposed in [13] and was used for signal classification in [72]. A robust signal classification algorithm was also proposed in [87], based on unsupervised learning.

Although the RL algorithms (such as Q-learning) may provide a suitable framework for autonomous unsupervised learning, their performance in partially observable, non-Markovian and multi-agent systems can be unsatisfactory [88]–[91]. Other types of learning mechanisms such as evolutionary learning [89], [92], learning by imitation, learning by instruction [93] and policy-gradient methods [90], [91] have been shown to outperform RL on certain problems under such conditions. For example, the policy-gradient approach has been shown to be more efficient in partially observable environments since it searches directly for optimal policies in the policy space, as we shall discuss later in this paper [90], [91].

Similarly, learning in multi-agent environments has been considered in recent years, especially when designing learning policies for CRNs. For example, [94] compared a cognitive network to a human society that exhibits both individual and group behaviors, and a strategic learning framework for cognitive networks was proposed in [95]. An evolutionary game framework was proposed in [96] to achieve adaptive learning in cognitive users during their strategic interactions. By taking into consideration the distributed nature of CRNs and the interactions among the CRs, optimal learning methods can be obtained based on cooperative schemes, which helps avoid the selfish behaviors of individual nodes in a CRN.

One of the main challenges of learning in distributed CRNs is the problem of action coordination [88]. To ensure optimal behavior, centralized policies may be applied to generate optimal joint actions for the whole network. However, centralized schemes are not always feasible in distributed networks. Hence, the aim of cognitive nodes in distributed networks is to apply decentralized policies that ensure near-optimal behavior while reducing the communication overhead among nodes. For example, a decentralized technique that was proposed in [3], [97] was based on the concept of *docitive networks*, from the Latin word *docere* (to teach), which establishes knowledge transfer (i.e. teaching) over the wireless medium [3]. The objective of docitive networks is to reduce the cognitive complexity, speed up the learning rate and generate better and more reliable decisions [3]. In a docitive network, radios teach each others by interchanging knowledge such that each node attempts to learn from a *more intelligent* node. The radios are not only supposed to teach end-results, but rather elements of the methods of getting there [3]. For example, in a docitive network, new upcoming radios can acquire certain policies from existing radios in the network. Of course, there will be communication overhead during the knowledge transfer process. However, as it is demonstrated in [3], [97], this overhead is compensated by the policy improvement achieved due to cooperative docitive behavior.

A. Purpose of this paper

This paper discusses the role of learning in CRs and emphasizes how crucial the autonomous learning ability in realizing a real CR device. We present a survey of the state-of-the-art achievements in applying machine learning techniques to CRs.

It is perhaps helpful to emphasize how this paper is different from other related survey papers. The most relevant is the survey of artificial intelligence for CRs provided in [98] which reviews several CR implementations that used the following artificial intelligence techniques: artificial neural networks

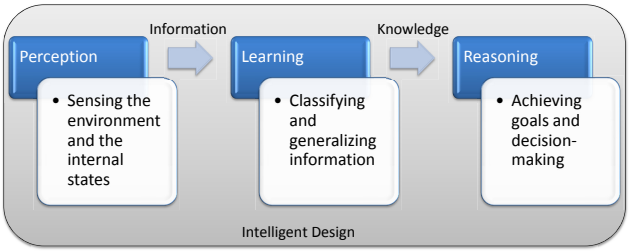


Fig. 1. An intelligent design can transform the acquired information into knowledge by learning.

(ANNs), metaheuristic algorithms, hidden Markov models (HMMs), rule-based reasoning (RBR), ontology-based reasoning (OBR), and case-based reasoning (CBR). To help readers better understand the design issues, two design examples are presented: one using an ANN and the other using CBR. The first example uses ordinary laboratory testing equipment to build a fast CR prototype. It also proves that, in general, an artificial intelligence technique (e.g., an ANN) can be chosen to accomplish complicated parameter optimization in the CR for a given channel state and application requirement. The second example builds upon the first example and develops a refined cognitive engine framework and process flow based on CBR.

Artificial intelligence includes several sub-categories such as knowledge representation and machine learning, machine perception, among others. In our survey, however, we focus on the special challenges that are encountered in applying machine learning techniques to CRs, given the importance of learning in CR applications, as we mentioned earlier. In particular, we provide in-depth discussions on the different types of learning paradigms in the two main categories: supervised learning and unsupervised learning. The machine learning techniques discussed in this paper include those that have been already proposed in the literature as well as those that might be reasonably applied to CRs in future. The advantages and limitations of these techniques are discussed to identify perhaps the most suitable learning methods in a particular context or in learning a particular task or an attribute. Moreover, we provide discussions on the centralized and decentralized learning techniques as well as the challenging machine learning problems in the non-Markovian environments.

B. Organization of the paper

This survey paper is organized as follows: Section II defines the learning problem in CRs and presents the different learning paradigms. Sections III and IV present the decision-making and feature classification problems, respectively. In Section V, we describe the learning problem in centralized and decentralized CRNs and we conclude the paper in Section VI.

II. Need of Learning in Cognitive Radios

A. Definition of the learning problem

A CR is defined to be "an intelligent wireless communication system that is aware of its environment and uses the methodology of understanding-by-building to learn from the environment and adapt to statistical variations in the input stimuli" [6]. As a result, a CR is expected to be intelligent by nature. It is capable of learning from its experience by interacting with its RF environment [5]. According to [99], learning should be an indispensable component of any intelligent system, which justifies it being designated a fundamental requirement of CRs.

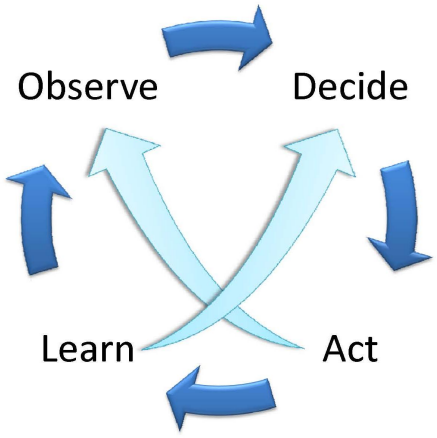


Fig. 2. The cognition cycle of an autonomous cognitive radio (referred to as the Radiobot) [5]. Decisions that drive Actions are made based on the Observations and Learnt knowledge. The impact of actions on the system performance and environment leads to new Learning. The Radiobot's new Observations are guided by this Learnt Knowledge of the effects of past Actions.

As identified in [99], there are three main conditions for intelligence: 1) Perception, 2) learning and 3) reasoning, as illustrated in Fig. 1. Perception is the ability of sensing the surrounding environment and the internal states to acquire information. Learning is the ability of transforming the acquired information into knowledge by using methodologies of classification and generalization of hypotheses. Finally, knowledge is used to achieve certain goals through reasoning. As a result, learning is at the core of any intelligent device including, in particular, CRs. It is the fundamental tool that allows a CR to acquire knowledge from its observed data.

In the followings, we discuss how the above three constituents of intelligence are built into CRs. First, *perception* can be achieved through the sensing measurements of the spectrum. This allows the CR to identify ongoing RF activities in its surrounding environment. After acquiring the sensing observations, the CR tries to *learn* from them in order to classify and organize the observations into suitable categories (knowledge). Finally, the *reasoning* ability allows the CR to use the knowledge acquired through learning to achieve its objectives. These characteristics were initially specified by Mitola in defining the so-called *cognition cycle* [1]. We illustrate in Fig. 2 an example of a simplified cognition cycle that was proposed in [5] for autonomous CRs, referred to as *Radiobots* [62]. Figure 2 shows that Radiobots can learn from their previous actions by observing their impact on the outcomes. The learning outcomes are then used to update, for example, the sensing (i.e. observation) and channel access (i.e. decision) policies in DSA applications [6], [16], [35], [38].

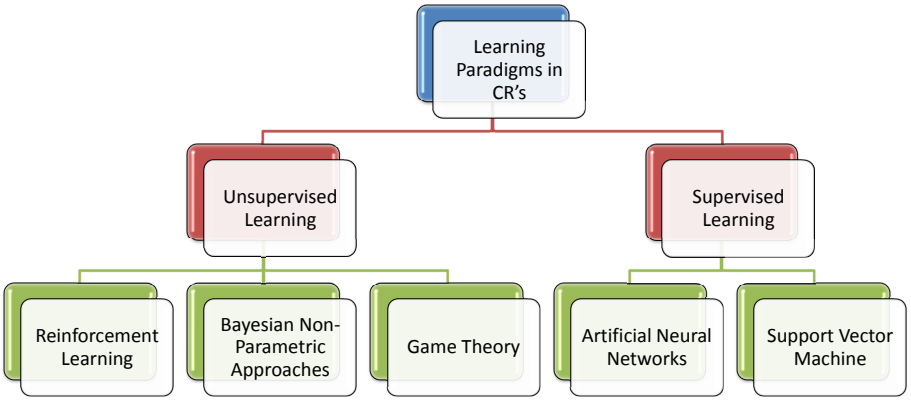


Fig. 3. Supervised and unsupervised learning approaches for cognitive radios.

B. Unique characteristics of cognitive radio learning problems

Although the term *cognitive radio* has been interpreted differently in various research communities [5], perhaps the most widely accepted definition is as a radio that can sense and adapt to its environment [2], [5], [6], [69]. The term *cognitive* implies *awareness*, *perception*, *reasoning* and *judgement*. As we already pointed out earlier, in order for a CR to derive reasoning and judgement from perception, it must possess the ability for learning [99]. Learning implies that the current actions should be based on past and current observations of the environment [100]. Thus, history plays a major role in the learning process of CRs.

Several learning problems are specific to CR applications due to the nature of the CRs and their operating RF environments. First, due to noisy observations and sensing errors, CRs can only obtain partial observations of their state variables. The learning problem is thus equivalent to a learning process in a partially observable environment and must be addressed accordingly.

Second, CRs in CRNs try to learn and optimize their behaviors simultaneously. Hence, the problem is naturally a multi-agent learning process. Furthermore, the desired learning policy may be based on either cooperative or non-cooperative schemes and each CR might have either full or partial knowledge of the actions of the other cognitive users in the network. In the case of partial observability, a CR might apply special learning algorithms to estimate the actions of the other nodes in the network before selecting its appropriate actions, as in, for example, [88].

Finally, autonomous learning methods are desired in order to enable CRs to learn on its own in an unknown RF environment. In contrast to licensed wireless users, a truly CR may be expected to operate in any available spectrum band, at any time and in any location [5]. Thus, a CR may not have any prior knowledge of the operating RF environment such as the noise or interference levels, noise distribution or user traffics. Instead, it should possess autonomous learning algorithms that may reveal the underlying nature of the environment and its components. This makes the unsupervised learning a perfect candidate for such learning problems in CR applications, as we shall point out throughout this survey paper.

To sum up, the three main characteristics that need to be considered when designing efficient learning algorithms for CRs are:

1) Learning in partially observable environments.
2) Multi-agent learning in distributed CRNs.
3) Autonomous learning in unknown RF environments.

A CR design that embeds the above capabilities will be able to operate efficiently and optimally in any RF environment.

C. Types of learning paradigms: Supervised versus unsupervised learning

Learning can be either supervised or unsupervised, as depicted in Fig. 3. Unsupervised learning may particularly be suitable for CRs operating in alien RF environments [5]. In this case, autonomous unsupervised learning algorithms permit exploring the environment characteristics and self-adapting actions accordingly without having any prior knowledge [5], [71]. However, if the CR has prior information about the environment, it might exploit this knowledge by using supervised learning techniques. For example, if certain signal waveform characteristics are known to the CR prior to its operation, training algorithms may help CRs to better detect signals with those characteristics.

In [93], the two categories of supervised and unsupervised learning are identified as learning by *instruction* and learning by *reinforcement*, respectively. A third learning regime is defined as the learning by *imitation* in which an agent learns by observing the actions of similar agents [93]. In [93], it was shown that the performance of a learning agent (learner) is influenced by its learning regime and its operating environment. Thus, to learn efficiently, a CR must adopt the best learning regime for a given learning problem, whether it is learning by *imitation*, by *reinforcement* or by *instruction* [93]. Of course, some learning regimes may not be applicable under certain circumstances. For example, in the absence of an instructor, the CR may not be able to learn by instruction and may have to resort to learning by reinforcement or imitation. An effective CR architecture is the one that can switch among different learning regimes depending on its requirements, the available information and the environment characteristics.

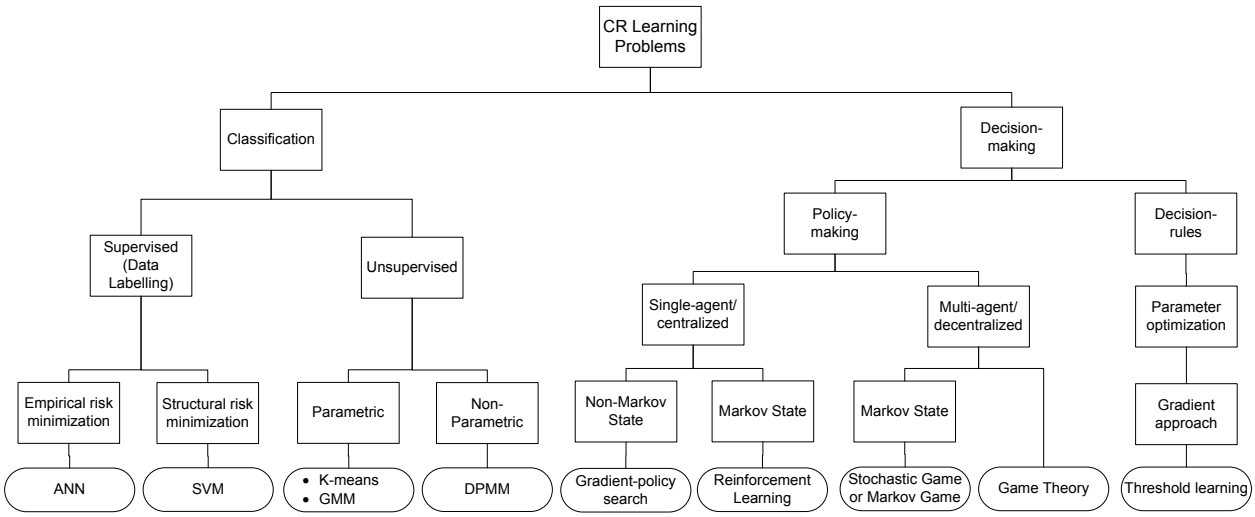


Fig. 4. Typical problems in cognitive radio and their corresponding learning algorithms.

D. Learning problems in cognitive radio

In this survey, we discuss several learning algorithms that can be used by CRs to achieve different goals. In order to obtain a better insight on the functions and similarities among the presented algorithms, we identify two main problem categories and show the learning algorithms under each category. The hierarchical organization of the learning algorithms and their dependence is illustrated in Fig. 4.

Referring to Fig. 4, we identify two main CR problems (or tasks) as:

1) Decision-making.
2) Feature classification.

These problems are general in a sense that they cover a wide range of CR tasks. For example, classification problems arise in spectrum sensing while decision-making problems arise in determining the spectrum sensing policy, power control or adaptive modulation.

The learning algorithms that are presented in this paper can be classified under the above two tasks, and can be applied under specific conditions, as illustrated in Fig. 4. For example, the classification algorithms can be split into two different categories: Supervised and unsupervised. Supervised algorithms require training with labeled data and include, among others, the ANN and SVM algorithms. The ANN algorithm is based on empirical risk minimization and does require prior knowledge of the observed process distribution, as opposed to structural models [101]–[103]. However, SVM algorithms, which are based on structural risk minimization, have shown superior performance, in particular for small training examples, since they avoid the problem of *overfitting* [101], [103].

For instance, consider a set of training data denoted as $\{(x_1, y_1), \cdots, (x_N, y_N)\}$ such that $x_i \in X$, $y_i \in Y$, $\forall i \in \{1, \cdots, N\}$. The objective of a supervised learning algorithm is to find a function $g : X \rightarrow Y$ that maximizes a certain score function [101]. In ANN, g is defined as the function that minimizes the empirical risk:

$$R(g) = R_{emp}(g) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, g(x_i)) \ , \qquad (1)$$

where $L : Y \times Y \rightarrow \mathbb{R}^+$ is a loss function. Hence, ANN algorithms find the function g that best fits the data. However, if the function space G includes too many candidates or the training set is not sufficiently large (i.e. small N), empirical risk minimization may lead to high variance and poor generalization, which is known as *overfitting*. In order to prevent overfitting, structural risk minimization can be used, which incorporates a regularization penalty to the optimization process [101]. This can be done by minimizing the following risk function:

$$R(g) = R_{emp}(g) + \lambda C(g) \ , \qquad (2)$$

where λ controls the bias/variance tradeoff and C is a penalty function [101].

In contrast with the supervised approaches, unsupervised classification algorithms do not require labeled training data and can be classified as being either parametric or non-parametric. Unsupervised parametric classifiers include the K-means and Gaussian mixture model (GMM) algorithms and require prior knowledge of the number of classes (or clusters). On the other hand, non-parametric unsupervised classifiers do not require prior knowledge of the number of clusters and can estimate this quantity from the observed data itself, for example using methods based on the Dirichlet process mixture model (DPMM) [72], [104], [105].

Decision-making is another major task that has been widely investigated in CR applications [17], [24]–[26], [35], [38], [77], [106]–[110]. Decision-making problems can in turn be split to policy-making and decision rules. Policy-making problems can be classified as either centralized or decentralized. In a policy-making problem, an agent determines its optimal set of actions over a certain time duration, thus defining an optimal policy (or an optimal strategy in game theory terminology). In a centralized scenario with a Markov state, RL algorithms can be used to obtain optimal solution to the

corresponding MDP, without prior knowledge of the transition probabilities [74], [76]. In non-Markov environments, optimal policies can be obtained based on gradient policy search algorithms which search directly for solutions in the policy space. On the other hand, for multi-agent scenarios, game theory is proposed as a solution that can capture the distributed nature of the environment and the interactions among users. With a Markov state assumption, the system can be modeled as a Markov game (or a stochastic game), while conventional game models can be used, otherwise. Note that learning algorithms can be applied to the game-theoretic models (such as the no-regret learning [111]–[113]) to arrive at equilibrium under uncertainty conditions.

Finally, decision rules form another class of decision-making problems which can be formulated as hypothesis testing problems for certain observation models. In the presence of uncertainty about the observation model, learning tools can be applied to implement a certain decision-rule. For example, the threshold-learning algorithm proposed in [72], [114] was used to optimize the threshold of the Neyman-Pearson test under uncertainty about the noise distribution.

In brief, we have identified two main classes of problems and have determined the conditions under which certain algorithms can be applied for these problems. For example, the DPMM algorithm can be applied for classification problems if the number of clusters is unknown, whereas the SVM may be better suited if labeled data is available for training.

The learning algorithms that are presented in this survey help to optimize the behavior of the learning agent (in particular the CR) under uncertainty conditions. For example, the RL leads to the optimal policy for MDPs [74] while game theory leads to Nash equilibrium, whenever it exists, of certain types of games [115]. The SVM algorithm optimizes the structural risk by finding a global minimum, whereas the ANN only leads to local minimum of the empirical risk [102], [103]. The DPMM is useful for non-parametric classification and converges to the stationary probability distribution of the Markov chain in the Markov-chain Monte-Carlo (MCMC) Gibbs sampling procedure [104], [116]. As a result, the proposed learning algorithms achieve certain optimality criterion within their application contexts.

III. Decision-making in cognitive radios

A. Centralized policy-making under Markov states: Reinforcement learning

Reinforcement learning is a technique that permits an agent to modify its behavior by interacting with its environment [74]. This type of learning can be used by agents to learn autonomously without supervision. In this case, the only source of knowledge is the feedback an agent receives from its environment after executing an action. Two main features characterize the RL: *trial-and-error* and *delayed reward*. By *trial-and-error* it is assumed that an agent does not have any prior knowledge about the environment, and executes actions blindly in order to *explore* the environment. The *delayed reward* is the feedback signal that an agent receives from the environment after executing each action. These rewards can be positive or negative quantities, telling *how good or bad* an action is. The agent's objective is to maximize these rewards by *exploiting* the system.

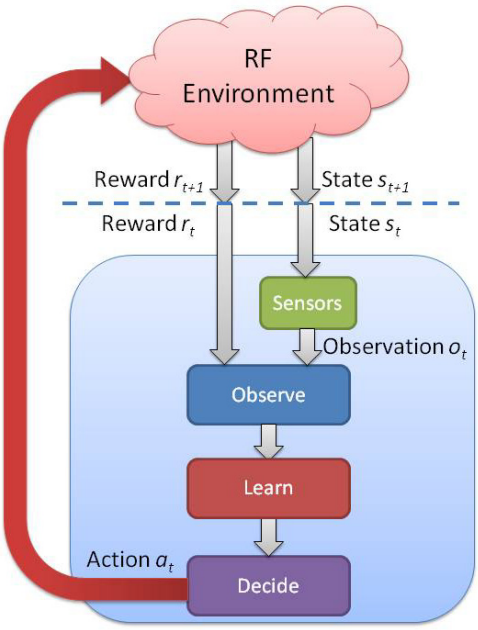


Fig. 5. The reinforcement learning cycle: At the beginning of each learning cycle, the agent receives a full or partial observation of the current state, as well as the accrued reward. By using the state observation and the reward value, the agent updates its policy (e.g. updating the Q-values) during the learning stage. Finally, during the decision stage, the agent selects a certain action according to the updated policy.

An RL-based cognition cycle for CRs was defined in [81], as illustrated in Fig. 5. It shows the interactions between the CR and its RF environment. The learning agent receives an observation o_t of the state s_t at time instant t. The observation is accompanied by a delayed reward $r_t(s_{t-1}, a_{t-1})$ representing the reward received at time t resulting from taking action a_{t-1} in state s_{t-1} at time $t-1$. The learning agent uses the observation o_t and the delayed reward $r_t(s_{t-1}, a_{t-1})$ to compute the action a_t that should be taken at time t. The action a_t results in a state transition from s_t to s_{t+1} and a delayed reward $r_{t+1}(s_t, a_t)$. It should be noted that here the learning agent is not passive and does not only observe the outcomes from the environment, but also affects the state of the system via its actions such that it might be able to drive the environment to a desired state that brings the highest reward to the agent.

1) RL for aggregate interference control: RL algorithms are applied under the assumption that the agent-environment interaction forms an MDP. An MDP is characterized by the following elements [76]:

- A set of *decision epochs* T including the point of times at which decisions are made. The time interval between decision epoch $t \in T$ and decision epoch $t+1 \in T$ is denoted as *period* t.
- A finite set $\mathcal{S}$ of states for the agent (i.e. secondary user).
- A finite set $\mathcal{A}$ of actions that are available to the agent. In particular, in each state $s \in \mathcal{S}$, a subset $\mathcal{A}_s \subseteq \mathcal{A}$ might be available.
- A non-negative function $p_t(s'|s,a)$ denoting the proba-

bility that the system is in state s' at time epoch $t+1$, when the decision-maker chooses action $a \in \mathcal{A}$ in state $s \in \mathcal{S}$ at time t. Note that, the subscript t might be dropped from $p_t(s'|s,a)$ if the system is stationary.

- A real-valued function $r_t^{MDP}(s,a)$ defined for state $s \in \mathcal{S}$ and action $a \in \mathcal{A}$ to denote the value at time t of the reward received in period t [76]. Note that, in RL literature, the reward function is usually defined as the delayed reward $r_{t+1}(s,a)$ that is obtained at time epoch $t+1$ after taking action a in state s at time t [74].

At each time epoch t, the agent observes the current state s and chooses an action a. An optimum policy maximizes the total expected rewards, which is usually discounted by a discount factor $\gamma \in [0,1)$ in case of an infinite time horizon. Thus, the objective is to find the optimal policy π that maximizes the expected *discounted return* [74]:

$$R(t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}(s_{t+k}, a_{t+k}) \ , \tag{3}$$

where s_t and a_t are, respectively, the state and action at time $t \in \mathbb{Z}$.

The optimal solution of an MDP can be obtained by using several methods such as the *value iteration* algorithm based on dynamic programming [76][1]. Given a certain policy π, the value of state $s \in \mathcal{S}$ is defined as the expected discounted return if the system starts in state s and follows policy π thereafter [74], [76]. This value function can be expressed as [74]:

$$V^{\pi}(s) = \mathbb{E}_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}(s_{t+k}, a_{t+k}) | s_t = s \right\} \ , \tag{4}$$

where $\mathbb{E}_{\pi}\{.\}$ denotes the expected value given that the agent follows policy π. Similarly, the value of taking action a in state s under a policy π is defined as the *action-value function* [74]:

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}(s_{t+k}, a_{t+k}) | s_t = s, a_t = a \right\} . \tag{5}$$

The value iteration algorithm finds an ε-optimal policy assuming stationary rewards and transition probabilities (i.e. $r_t(s,a) = r(s,a)$ and $p_t(s'|s,a) = p(s'|s,a)$). The algorithm initializes a $v^0(s)$ for each $s \in \mathcal{S}$ arbitrarily and iteratively updates $v^n(s)$ (where $v^n(s)$ is the estimated value of state s after the n-th iteration) for each $s \in \mathcal{S}$ as follows [76]:

$$v^{n+1}(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{j \in \mathcal{S}} p(j|s,a) v^n(j) \right\} \ . \tag{6}$$

The algorithm stops when $\|v^{n+1} - v^n\| < \varepsilon \frac{1-\gamma}{2\gamma}$ and the ε-optimal decision $d_\epsilon(s)$ of each state $s \in \mathcal{S}$ is defined as:

$$d_\epsilon(s) = \arg\max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{j \in \mathcal{S}} p(j|s,a) v^{n+1}(j) \right\} \ . \tag{7}$$

[1]There are other algorithms that can be applied to find the optimal policy of an MDP such as *policy iteration* and *linear programming* methods. Interested readers are referred to [76] for additional information regarding these methods.

Obviously, the *value iteration* algorithm requires explicit knowledge of the transition probability $p(s'|s,a)$. On the other hand, an RL algorithm, referred to as the Q-learning, was proposed by Watkins in 1989 [117] to solve the MDP problem without knowledge of the transition probabilities and has been recently applied to CRs [38], [77], [82], [118]. The Q-learning algorithm is one of the important *temporal difference* (TD) methods [74], [117]. It has been shown to converge to the optimal policy when applied to single agent MDP models (i.e. centralized control) in [117] and [74]. However, it can also generate satisfactory near-optimal solutions even for decentralized partially observable MDPs (DEC-POMDPs), as shown in [77]. The *one-step* Q-learning is defined as follows:

$$\begin{aligned} Q(s_t,a_t) &\leftarrow (1-\alpha) Q(s_t,a_t) + \\ &+ \alpha \left[r_{t+1}(s_t,a_t) + \gamma \max_a Q(s_{t+1},a) \right] . \end{aligned} \tag{8}$$

The learned action-value function, Q in (8), directly approximates the optimal action-value function Q^* [74]. However, it is required that all state-action pairs need to be continuously updated in order to guarantee *correct convergence* to Q^*. This can be achieved by applying an ε-greedy policy that ensures that all state-action pairs are updated with a non-zero probability, thus leading to an optimal policy [74]. If the system is in state $s \in \mathcal{S}$, the ε-greedy policy selects action $a^*(s)$ such that:

$$a^*(s) = \begin{cases} \arg\max_{a \in \mathcal{A}} Q(s,a) & \text{, with Pr} = 1-\varepsilon \\ \sim U(\mathcal{A}) & \text{, with Pr} = \varepsilon \end{cases} , \tag{9}$$

where $U(\mathcal{A})$ is the discrete uniform probability distribution over the set of actions $\mathcal{A}$.

In [77], the authors applied the Q-learning to achieve interference control in a cognitive network. The problem setup of [77] is illustrated in Fig. 6 in which multiple IEEE 802.22 WRAN cells are deployed around a Digital TV (DTV) cell such that the aggregated interference caused by the secondary networks to the DTV network is below a certain threshold. In this scenario, the CR (agents) constitutes a distributed network and each radio tries to determine how much power it can transmit so that the aggregated interference on the primary receivers does not exceed a certain threshold level.

In this system, the secondary base stations form the learning agents that are responsible for identifying the current environment state, selecting the action based on the Q-learning methodology and executing it. The state of the i-th WRAN network at time t consists of three components and is defined as [77]:

$$s_t^i = \{ I_t^i, d_t^i, p_t^i \} \ , \tag{10}$$

where I_t^i is a binary indicator specifying whether the secondary network generates interference to the primary network above or below the specified threshold, d_t^i denotes an estimate of the distance between the secondary user and the interference contour, and p_t^i denotes the current power at which the secondary user i is transmitting. In the case of full state observability, the secondary user has complete knowledge of the state of the environment. However, in a partially observable environment, the agent i has only partial information of the actual state and uses a belief vector to represent the probability

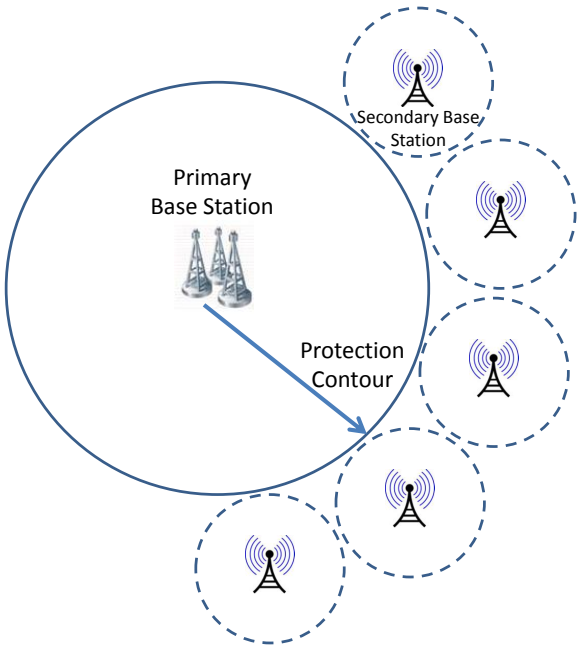


Fig. 6. System model of [77] which is formed of a Digital TV (DTV) cell and multiple WRAN cells.

distribution of the state values. In this case, the randomness in s_t^i is only related to the parameter I_t^i which is characterized by two elements $\mathcal{B} = \{b(1), b(2)\}$, i.e. the values of the probability mass function (pmf) of I_t^i.

The set of possible actions is the set $\mathcal{P}$ of power levels that the secondary base station can assign to the i-th user. The cost c_t^i denotes the immediate reward incurred due to the assignment of action a in state s and is defined as:

$$c = \left(SINR_t^i - SINR_{Th}\right)^2 , \tag{11}$$

where $SINR_t^i$ is the instantaneous SINR at the control point of WRAN cell i whereas $SINR_{Th}$ is the maximum value of SINR that can be perceived by primary receivers [77].

By applying the Q-learning algorithm, the results in [77] showed that it can control the interference to the primary receivers, even in the case of partial state observability. Thus, the network can achieve equilibrium in a completely distributed way without the intervention of centralized controllers. By using the past experience and by interacting with the environment, the decision-makers can achieve self-adaptation progressively in real-time. Note that, a learning phase is required to acquire the optimal/suboptimal policy. However, once this policy is reached, the multi-agent system takes only one iteration to reach the optimal power configuration, starting at any initial state [77].

2) Docition in cognitive radio networks: As we have discussed above, the decentralized decision-makers of a CRN require a learning phase before acquiring an optimal/suboptimal policy. The learning phase will cause delays in the adaptation process since each agent has to learn individually from scratch [3], [97]. In an attempt to resolve this problem, the authors in [3], [97] proposed a timely solution to enhance the learning process in decentralized CRNs by allowing efficient cooperation among the learning agents. They proposed docitive algorithms aimed at reducing the complexity of cooperative learning in decentralized networks. Docitive algorithms are based on the concept of knowledge sharing in which different nodes try to teach each other by exchanging their learning skills. The learning skills do not simply consist of certain end observations or decisions. Cognitive nodes in a docitive system can learn certain policies and learning techniques from other nodes that have demonstrated superior performance.

In particular, the authors in [3], [97] applied the docition paradigm to the same problem of aggregated interference control that was presented in [77] and described above. The authors compared the performance of the CRN under both docitive and non-docitive policies and showed that docition leads to superior performance in terms of speed of convergence and precision (i.e. oscillations around the target SINR) [3].

In [97], the authors proposed three different docitive approaches that can be applied in CRNs:

- Startup docition: Docitive radios teach their policies to any newcoming radios joining the network. In practice, this can be achieved by supplying the Q-table of the incumbent radios to newcomers. Hence, new radios do not have to learn from scratch, but instead can use the learnt policies of existing radios to speed-up their learning process. Note that, newcomer radios learn independently after initializing their Q-tables. However, the information and policy exchange among radios is useful at the beginning of the learning process due to high correlation among the different learning nodes in the cognitive network.
- Adaptive docition: According to this technique, CRs share policies based on performances. The learning nodes share information about the performance parameters of their learning processes such as variance of the oscillations with respect to the target and speed of convergence. Based on this information, cognitive nodes can learn from neighboring nodes that are performing better.
- Iterative docition: Docitive radios periodically share parts of their policies based on the reliability of their expert knowledge. Expert nodes exchange rows of the Q-table corresponding to the states that have been previously visited.

By comparing the docitive algorithms with the independent learning case described in [77], the results in [97] showed that docitive algorithms achieve faster convergence and more accurate results. Furthermore, compared to other docitive algorithms, iterative docitive algorithms have shown superior performance [97].

B. Centralized policy-making with non-Markovian states: Gradient-policy search

While RL and *value-iteration* methods [74], [76] can lead to optimal policies for the MDP problem, their performance in non-Markovian environments remains questionable [90], [91]. Hence, the authors in [89]–[91] proposed the *policy-search* approach as an alternative solution method for non-Markovian learning tasks. Policy-search algorithms directly

look for optimal policies in the policy space itself, without having to estimate the actual states of the systems [90], [91]. In particular, by adopting policy gradient algorithms, the policy vector can be updated to reach an optimal solution (or a local optimum) in non-Markovian environments.

The value-iteration approach has several other limitations as well: First, it is restricted to deterministic policies. Second, any small changes in the estimated value of an action can cause that action to be, or not to be selected [90]. This would affect the optimality of the resulting policy since optimal actions might be eliminated due to an underestimation of their value functions.

On the other hand, the gradient-policy approach has shown promising results, for example, in robotics applications [119], [120]. Compared to value-iteration methods, the gradient-policy approach requires fewer parameters in the learning process and can be applied in model-free setups not requiring prefect knowledge of the controlled system.

The policy-search approach can be illustrated by the following overview of policy-gradient algorithms from [91]. We consider a class of stochastic policies that are parameterized by $\theta \in \mathbb{R}^K$. By computing the gradient with respect to θ of the average reward, the policy could be improved by adjusting the parameters in the gradient direction. To be concrete, assume $r(X)$ to be a reward function that depends on a random variable X. Let $q(\theta, x)$ be the probability of the event $\{X = x\}$. The gradient with respect to θ of the expected performance $\eta(\theta) = \mathbb{E}\{r(X)\}$ can be expressed as:

$$\nabla\eta(\theta) = \mathbb{E}\left\{ r(X)\frac{\nabla q(\theta, x)}{q(\theta, x)} \right\} . \tag{12}$$

An unbiased estimate of the gradient can be obtained via simulation by generating N independent identically distributed (i.i.d.) random variables $X_1, \cdots, X_N$ that are distributed according to $q(\theta, x)$. The unbiased estimate of $\nabla\eta(\theta)$ is thus expressed as:

$$\hat{\nabla}\eta(\theta) = \frac{1}{N}\sum_{i=1}^{N} r(X_i)\frac{\nabla q(\theta, X_i)}{q(\theta, X_i)} . \tag{13}$$

By the law of large numbers, $\hat{\nabla}\eta(\theta) \rightarrow \nabla\eta(\theta)$ with probability one. Note that the quantity $\frac{\nabla q(\theta, X_i)}{q(\theta, X_i)}$ is referred to as the *likelihood ratio* or the *score function*. By having an estimate of the reward gradient, the policy parameter $\theta \in \mathbb{R}^K$ can be updated by following the gradient direction, such that:

$$\theta_{k+1} \leftarrow \theta_k + \alpha_k \nabla\eta(\theta) , \tag{14}$$

for some step size $\alpha_k > 0$.

Authors in [119], [120] identify two major steps when performing policy gradient methods:

1) A policy evaluation step in which an estimate of the gradient $\nabla\eta(\theta)$ of the expected return $\eta(\theta)$ is obtained, given a certain policy π_θ.
2) A policy improvement step which updates the policy parameter θ through steepest gradient ascent $\theta_{k+1} = \theta_k + \alpha_k \nabla\eta(\theta)$.

Note that, estimating the gradient $\nabla\eta(\theta)$ is not straightforward, especially in the absence of simulators that generate the X_i's. To resolve this problem, special algorithms can be designed to obtain reasonable approximations of the gradient. Indeed, several approaches have been proposed to estimate the gradient policy vector, mainly in robotics applications [119], [120]. Three different approaches have been considered in [120] for policy gradient estimation:

1) Finite difference (FD) methods.
2) Vanilla policy gradient (VPG) methods.
3) Natural policy gradient (NG) methods.

Finite difference (FD) methods, originally used in stochastic simulations literature, are among the oldest policy gradient approaches. The idea is based on changing the current policy parameter θ_k by small perturbations $\delta\theta_i$ and computing $\delta\eta_i = \eta(\theta_k + \delta\theta_i) - \eta(\theta_k)$. The policy gradient $\nabla\eta(\theta)$ can be thus estimated as:

$$\mathbf{g}_{FD} = \left(\boldsymbol{\Delta\Theta}^T\boldsymbol{\Delta\Theta}\right)^{-1}\boldsymbol{\Delta\Theta}\boldsymbol{\Delta\eta} , \tag{15}$$

where $\boldsymbol{\Delta\Theta} = [\delta\theta_1, \cdots, \delta\theta_I]^T$, $\boldsymbol{\Delta\eta} = [\delta\eta_1, \cdots, \delta\eta_I]^T$ and I is the number of samples [119], [120]. Advantages of this approach is that it is straightforward to implement and does not introduce significant noise to the system during exploration. However, the gradient estimate can be very sensitive to perturbations (i.e. $\delta\theta_i$) which may lead to bad results [120].

Instead of perturbing the parameter θ_k of a deterministic policy $u = \pi(x)$ (with u being the action and x being the state), the VPG approach assumes a stochastic policy $u \sim \pi(u|x)$ and obtains an unbiased gradient estimate [120]. However, in using the VPG method, the variance of the gradient estimate depends on the squared average magnitude of the reward, which can be very large. In addition, the convergence of the VPG to the optimal solution can be very slow, even with an optimal baseline [120]. The NG approach which leads to fast policy gradient algorithms can alleviate this problem. Natural gradient approaches use the Fisher information $F(\theta)$ to characterize the information about the policy parameters θ that is contained in the observed path τ [120]. A path (or a trajectory) $\tau = [x_{0:H}, u_{0:H}]$ is defined as the sequence of states and actions, where H denotes the horizon which can be infinite [119]. Thus, the Fisher information $F(\theta)$ can be expressed as:

$$F(\theta) = \mathbb{E}\left\{\nabla_\theta \log p(\tau|\theta) \nabla_\theta \log p(\tau|\theta)^T\right\} , \tag{16}$$

where $p(\tau|\theta)$ is the probability of trajectory τ, given certain policy parameter θ. For a given policy change $\delta\theta$, there is an information loss of $l_\theta(\delta\theta) \approx \delta\theta^T F(\theta)\delta\theta$, which can also be seen as the change in path distribution $p(\tau|\theta)$. By searching for the policy change $\delta\theta$ that maximizes the expected return $\eta(\theta + \delta\theta)$ for a constant information loss $l_\theta(\delta\theta) \approx \varepsilon$, the algorithms searches for the highest return value on an ellipse around the current parameter θ and then goes in the direction of the highest values. More formally, the direction of the steepest ascent on the ellipse around θ can be expressed as [120]:

$$\delta\theta = \arg\max_{\delta\theta \text{ s.t. } l_\theta(\delta\theta)=\varepsilon} \delta\theta^T \nabla_\theta \eta(\theta) = F^{-1}(\theta)\nabla_\theta\eta(\theta) . \tag{17}$$

This algorithm is further explained in [120] and can be easily implemented based on the Natural Actor-Critic algorithms [120].

By comparing the above three approaches, the authors in [120] showed that NG and VPG methods are considerably faster and result in better performance, compared to FD. However, FD has the advantage of being simpler and applicable in more general situations.

C. Decentralized policy-making: Game Theory

Game theory [121] presents a suitable platform for modeling rational behavior among CRs in CRNs. There is a rich literature on game theoretic techniques in CR, as can be found in [11], [122]–[132]. A survey on game theoretic approaches for multiple access wireless systems can be found in [115].

Game theory [121] is a mathematical tool that attempts to implement the behavior of rational entities in an environment of conflict. This branch of mathematics has primarily been popular in economics, and has later found its way into biology, political science, engineering and philosophy [115]. In wireless communications, game theory has been applied to data communication networking, in particular, to model and analyze routing and resource allocation in competitive environments.

A game model consists of several rational entities that are denoted as the players. Assuming a game model $\mathbb{G} = (\mathcal{N}, (\mathcal{A}_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}})$, where $\mathcal{N} = \{1, \cdots, N\}$ denotes the set of N players and each player $i \in \mathcal{N}$ has a set $\mathcal{A}_i$ of available actions and a utility function U_i. Let $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$ be the set of strategy profiles of all players. In general, the utility function of an individual player $i \in \mathcal{N}$ depends on the actions taken by all the players involved in the game and is denoted as $U_i(a_i, a_{-i})$, where $a_i \in \mathcal{A}_i$ is an action (or strategy) of player i and $a_{-i} \in \mathcal{A}_{-i}$ is a strategy profile of all players except player i. Each player selects its strategy in order to maximize its utility function. A Nash equilibrium of a game is defined as a point at which the utility function of each player does not increase if the player deviates from that point, given that all the other players' actions are fixed. Formally, a strategy profile $(a_1^*, \cdots, a_N^*) \in \mathcal{A}$ is a Nash equilibrium if [112]:

$$U_i(a_i^*, a_{-i}) \geq U_i(a_i', a_{-i}), \forall i \in \mathcal{N}, \forall a_i' \in \mathcal{A}_i \ . \qquad (18)$$

A key advantage of applying game theoretic solutions to CR protocols is in reducing the complexity of adaptation algorithms in large cognitive networks. While optimal centralized control can be computationally prohibitive in most CRNs, due to communication overhead and algorithm complexity, game theory presents a distributed platform to handle such situations [98]. Another justification for applying game theoretic approaches to CRs is the assumed cognition in the CR behavior, which induces *rationality* among CRs, similar to the players in a game.

1) Game Theoretic Approaches: There are two major game theoretic approaches that can be used to model the behavior of nodes in a wireless medium: Cooperative and non-cooperative games. In a non-cooperative game, the players make rational decisions considering only their individual payoff. In a cooperative game, however, players are grouped together and establish an enforceable agreement in their group [115].

A non-cooperative game can be classified as either a complete or an incomplete information game. In a complete information game, each player can observe the information of other players such as their payoffs and their strategies. On the other hand, in an incomplete information game, this information is not available to other players. A game with incomplete information can be modeled as a Bayesian game in which the game outcomes can be estimated based on Bayesian analysis. A Bayesian Nash equilibrium is defined for the Bayesian game, similar to the Nash equilibrium in the complete information game [115].

In addition, a game can also be classified as either static or dynamic. In a static game, each player takes its actions without knowledge of the strategies taken by the other players. This is denoted as a one-shot game which ends when actions of all players are taken and payoffs are received. In a dynamic game, however, a player selects an action in the current stage based on the knowledge of the actions taken by the other players in the current or previous stages. A dynamic game is also called a sequential game since it consists of a sequence of repeated static games. The common equilibrium solution in dynamic games is the subgame perfect Nash equilibrium which represents a Nash equilibrium of every subgame in the original game [115].

2) Applications of Game Theory to Cognitive Radios: Several types of games have been adapted to model different situations in CRNs [98]. For example, supermodular games (the games having the following important and useful property: there exists at least one pure strategy Nash equilibrium) have been used for distributed power control in [133], [134] and for rate adaptation in [135]. Repeated games were applied for DSA by multiple secondary users that share the same spectrum hole in [136]. In this context, repeated games are useful in building reputations and applying punishments in order to reinforce a certain desired outcome. The Stackelberg game model can be used as a model for implementing CR behavior in cooperative spectrum leasing where the primary users act as the game-leaders and secondary cognitive users as the followers [50].

Auctions are one of the most popular methods used for selling a variety of items, ranging from antiques to wireless spectrum. In auction games the players are the buyers who must select the appropriate bidding strategy in order to maximize their perceived utility (i.e., the value of the acquired items minus the payment to the seller). The concept of auction games has successfully been applied to cooperative dynamic spectrum leasing (DSL) in [37], [137], as well as to spectrum allocation problems in [138]. The basics of the auction games and the open challenges of applying auction games to the field of spectrum management are discussed in [139].

Stochastic games (or Markov games) can be used to model the greedy selfish behavior of CRs in a CRN, where CRs try to learn their best response and improve their strategies over time [140]. In the context of CRs, stochastic games are dynamic, competitive games with probabilistic actions played by secondary spectrum users. The game is played in a sequence of stages. At the beginning of each stage, the game is in a certain state. The secondary users choose their actions, and each secondary user receives a reward that

depends on both its current state and its selected actions. The game then moves to the next stage having a new state with a certain probability, which depends on the previous state as well as the actions selected by the secondary users. The process continues for a finite or infinite number of stages. The stochastic games are generalizations of repeated games that have only a single state.

3) Learning in Game Theoretic Models: There are several learning algorithms that have been proposed to estimate unknown parameters in a game model (e.g. other players' strategies, environment states, etc.). In particular, no-regret learning allows initially uninformed players to acquire knowledge about their environment state in a repeated game [111]. This algorithm does not require prior knowledge of the number of players nor the strategies of other players. Instead, each player will learn a better strategy based on the rewards obtained from playing each of its strategies [111].

The concept of regret is related to the benefit a player feels after taking a particular action, compared to other possible actions. This can be computed as the average reward the player gets from a particular action, averaged over all other possible actions that could be taken instead of that particular action. Actions resulting in lower regret are updated with higher weights and are thus selected more frequently [111]. In general, no-regret learning algorithms help players to choose their policies when they do not know the other players' actions. Furthermore, no-regret learning can adapt to a dynamic environment with little system overhead [111].

No-regret learning was applied in [111] to allow a CR to update both its transmission power and frequencies simultaneously. In [113], it was used to detect malicious nodes in spectrum sensing whereas in [112] no-regret learning was used to achieve a correlated equilibrium in opportunistic spectrum access for CRs. Assuming the game model $\mathbb{G} = (\mathcal{N}, (\mathcal{A}_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}})$ defined above, in a correlated equilibrium, a strategy profile $(a_1, \cdots, a_N) \in \mathcal{A}$ is chosen randomly according to a certain probability distribution p [112]. A probability distribution p is a correlated strategy, if and only if, for all $i \in \mathcal{N}$, $a_i \in \mathcal{A}_i$, $a_{-i} \in \mathcal{A}_{-i}$ [112]:

$$\sum_{a_{-i}\in\mathcal{A}_{-i}} p(a_i, a_{-i}) \left[U_i(a_i', a_{-i}) - U_i(a_i, a_{-i})\right] \leq 0, \forall a_i' \in \mathcal{A}_i \ . \tag{19}$$

Note that, every Nash equilibrium is a correlated equilibrium and Nash equilibria correspond to the special case where $p(a_i, a_{-i})$ is a product of each individual player's probability for different actions, i.e. the play of the different players is independent [112]. Compared to the non-cooperative Nash equilibrium, the correlated equilibrium in [112] was shown to achieve better performance and fairness.

Recently, [141] proposed a game-theoretic stochastic learning solution for opportunistic spectrum access when the channel availability statistics and the number of secondary users are unknown a priori. This model attempts to resolve non-feasible opportunistic spectrum access solution which requires prior knowledge of the environment and the actions taken by the other users. By applying the stochastic learning solution in [141], the communication overhead among the CR users is reduced. Furthermore, the model in [141] provides an alternative solution to opportunistic spectrum access schemes proposed in [107], [108] that do not consider the interactions among multiple secondary users in a partially observable MDP (POMDP) framework [141].

Thus, learning in a game theoretic framework can help CRs to adapt to environment variations given a certain uncertainty about the other users' strategies. Therefore, it provides a potential solution for multi-agent learning problems under partial observability assumptions.

D. Decision rules under uncertainty: Threshold-learning

A CR may be implemented on a mobile device that changes location over time and switches transmissions among several channels. This mobility and multi-band/multi-channels operability may pose a major challenge for CRs in adapting to their RF environments. A CR may encounter different noise or interference levels when switching between different bands or when moving from one place to another. Hence, the operating parameters (e.g. test thresholds and sampling rate) of CRs need to be adapted with respect to each particular situation. Moreover, CRs may be operating in unknown RF environments and may not have perfect knowledge of the characteristics of the other existing primary or secondary signals, requiring special learning algorithms to allow the CR to explore and adapt to its surrounding environment. In this context, special types of learning can be applied to directly learn the optimal values of certain design and operation parameters.

Threshold learning presents a technique that permits such dynamic adaptation of operating parameters to satisfy the performance requirements, while continuously learning from the past experience. By assessing the effect of previous parameter values on the system performance, the learning algorithm optimizes the parameters values to ensure a desired performance. For example, in considering energy detection, after measuring the energy levels at each frequency, a CR decides on the occupancy of a certain frequency band by comparing the measured energy levels to a certain threshold. The threshold levels are usually designed based on Neyman-Pearson tests in order to maximize the detection probability of primary signals, while satisfying a constraint on the false alarm. However, in such tests, the optimal threshold depends on the noise level. An erroneous estimation of the noise level might cause sub-optimal behavior and violation of the operation constraints (for example, exceeding a tolerable collision probability with primary users). In this case, and in the absence of perfect knowledge about the noise levels, threshold-learning algorithms can be devised to learn the optimal threshold values. Given each choice of a threshold, the resulting false alarm rate determines how the test threshold should be regulated to achieve a desired false alarm probability. An example of application of threshold learning can be found in [75] where a threshold learning algorithm was derived for optimizing spectrum sensing in CRs. The resulting algorithm was shown to converge to the optimal threshold that satisfies a given false alarm probability.

IV. Feature Classification in Cognitive Radios

A. Non-parametric unsupervised classification: The Dirichlet Process Mixture Model

A major challenge an autonomous CR can face is the lack of knowledge about the surrounding RF environment [5], in particular, when operating in the presence of unknown primary signals. Even in such situations, a CR is expected to be able to adapt to its environment while satisfying certain requirements. For example, in DSA, a CR must not exceed a certain collision probability with primary users. For this reason, a CR should be equipped with the ability to autonomously explore its surrounding environment and to make decisions about the primary activity based on the observed data. In particular, a CR must be able to extract knowledge concerning the statistics of the primary signals based on measurements [5], [72]. This makes unsupervised learning an appealing approach for CRs in this context. In the following, we may explore a Dirichlet process prior based [142], [143] technique as a framework for such non-parametric learning and point out its potentials and limitations. The Dirichlet process prior based techniques are considered as unsupervised learning methods since they make few assumptions about the distribution from which the data is drawn [104], as can been seen in the following discussion.

A Dirichlet process $DP(\alpha_0, G_0)$ is defined to be the distribution of a random probability measure G that is defined over a measurable space $(\Theta, \mathcal{B})$, such that, for any finite measurable partition $(A_1, \cdots, A_r)$ of Θ, the random vector $(G(A_1), \cdots, G(A_r))$ is distributed as a finite dimensional Dirichlet distribution with parameters $(\alpha_0 G_0(A_1), \cdots, \alpha_0 G_0(A_r))$, where $\alpha_0 > 0$ [104]. We denote:

$$(G(A_1), \cdots, G(A_r)) \sim Dir(\alpha_0 G_0(A_1), \cdots, \alpha_0 G_0(A_r)) \ , \tag{20}$$

where $G \sim DP(\alpha_0, G_0)$, denotes that the probability measure G is drawn from the Dirichlet process $DP(\alpha_0, G_0)$. In other words, G is a *random probability measure* whose distribution is given by the Dirichlet process $DP(\alpha_0, G_0)$ [104].

1) Construction of the Dirichlet process: Teh [104] describes several ways of constructing the Dirichlet process. A first method is a direct approach that constructs the random probability distribution G based on the *stick-breaking* method. The *stick-breaking* construction of G can be summarized as follows [104]:

1) Generate independent i.i.d. sequences $\{\pi'_k\}_{k=1}^{\infty}$ and $\{\phi_k\}_{k=1}^{\infty}$ such that

$$\begin{cases} \pi'_k|\alpha_0, G_0 & \sim & Beta(1, \alpha_0) \\ \phi_k|\alpha_0, G_0 & \sim & G_0 \end{cases} \ , \tag{21}$$

where Beta(a, b) is the beta distribution whose probability density function (pdf) is given by $f(x, a, b) = \frac{x^{a-1}(1-x)^{b-1}}{\int_0^1 u^{a-1}(1-u)^{b-1} du}$.

2) Define $\pi_k = \pi'_k \prod_{l=1}^{k-1}(1 - \pi'_l)$. We can write $\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots) \sim GEM(\alpha_0)$, where GEM stands for Griffiths, Engen and McCloskey [104]. The $GEM(\alpha)$ process generates the vector $\boldsymbol{\pi}$ as described above, given a parameter α_0 in (21).

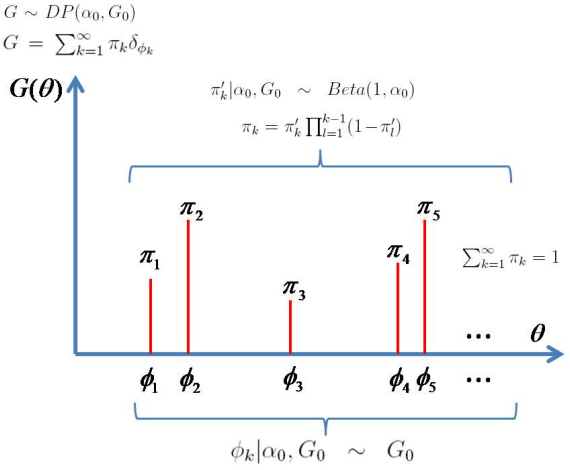


Fig. 7. One realization of the Dirichlet process.

3) Define $G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}$, where δ_ϕ is a probability measure concentrated at ϕ (and $\sum_{k=1}^{\infty} \pi_k = 1$).

In the above construction G is a random probability measure distributed according to $DP(\alpha_0, G_0)$. The randomness in G stems from the random nature of both the weights π_k and the weights positions ϕ_k. A sample distribution G of a Dirichlet process is illustrated in Fig. 7, using the steps described above in the *stick-breaking* method. Since G has an infinite discrete support (i.e. $\{\phi_k\}_{k=1}^{\infty}$), this makes it a suitable candidate for non-parametric Bayesian classification problems in which the number of clusters is unknown *a priori* (i.e. allowing for infinite number of clusters), with the infinite discrete support (i.e. $\{\phi_k\}_{k=1}^{\infty}$ being the set of clusters. However, due to the infinite sum in G, it may not be practical to construct G directly by using this approach in many applications. An alternative approach to construct G is by using either the Polya urn model [143] or the Chinese Restaurant Process (CRP) [144]. The CRP is a discrete-time stochastic process. A typical example of this process can be described by a Chinese restaurant with infinitely many tables and each table (cluster) having infinite capacity. Each customer (feature point) that arrives at the restaurant (RF spectrum) will choose a table with a probability proportional to the number of customers on that table. It may also choose a new table with a certain fixed probability.

A second approach to constructing a Dirichlet process does not define G explicitly. Instead, it characterizes the distribution of the drawings θ of G. Note that G is discrete with probability 1. For example, the Polya urn model [143] does not construct G directly, but it characterizes the draws from G. Let $\theta_1, \theta_2, \cdots$ be i.i.d. random variables distributed according to G. These random variables are independent, given G. However, if G is integrated out, $\theta_1, \theta_2, \cdots$ are no more conditionally independent and they can be characterized as:

$$\theta_i|\{\theta_j\}_{j=1}^{i-1}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{m_k}{i - 1 + \alpha_0} \delta_{\phi_k} + \frac{\alpha_0}{i - 1 + \alpha_0} G_0 \ , \tag{22}$$

where $\{\phi_k\}_{k=1}^K$ are the K distinct values of θ_i's and m_k is the number of values of θ_i that are equal to ϕ_k. Note that this conditional distribution is not necessarily discrete since G_0 might be a continuous distribution (in contrast with G which is discrete with probability 1). The θ_i's that are drawn from G exhibit a clustering behavior since a certain value of θ_i is most likely to reoccur with a nonnegative probability (due to the point mass functions in the conditional distribution). Moreover, the number of distinct θ_i values is infinite, in general, since there is a nonnegative probability that the new θ_i value is distinct from the previous $\theta_1, \cdots, \theta_{i-1}$. This conforms with the definition of G as a pmf over an infinite discrete set. Since θ_i's are distributed according to G, given G, we denote:

$$\theta_i | G \sim G \ . \tag{23}$$

2) Dirichlet Process Mixture Model: The Dirichlet process makes a perfect candidate for non-parametric classification problems through the DPMM. The DPMM imposes a non-parametric prior on the parameters of the mixture model [104]. The DPMM can be defined as follows:

$$\left\{ \begin{array}{lcl} G & \sim & DP(\alpha_0, G_0) \\ \theta_i | G & \sim & G \\ y_i | \theta_i & \sim & f(\theta_i) \end{array} \right. , \tag{24}$$

where θ_i's denote the mixture components and the y_i is drawn according to this mixture model with a density function f given a certain mixture component θ_i.

3) Data clustering based on the DPMM and the Gibbs sampling: Consider a sequence of observations $\{y_i\}_{i=1}^N$ and assume that these observations are drawn from a mixture model. If the number of mixture components is unknown, it is reasonable to assume a non-parametric model, such as the DPMM. Thus, the mixture components θ_i are drawn from $G \sim DP(\alpha_0, G_0)$, where G can be expressed as $G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}$, ϕ_k's are the unique values of θ_i, and π_k are their corresponding probabilities. Denote $\mathbf{y} = (y_1, \cdots, y_N)$.

The problem is to estimate the mixture component $\hat{\theta}_i$ for each observation y_i, for all $i \in \{1, \cdots, N\}$. This can be achieved by applying the Gibbs sampling method proposed in [116] which has been applied for various unsupervised clustering problems, such as speaker clustering problem in [145]. The Gibbs sampling is a technique for generating random variables from a (marginal) distribution indirectly, without having to calculate the density. As a result, by using te Gibbs sampling, one can avoid difficult calculations, replacing them instead with a sequence of easier calculations. Although the roots of the Gibbs sampling can be traced back to at least Metropolis et al. [146], the Gibbs sampling perhaps became more popular after the paper of Geman and Geman [147], who studied image-processing models.

In the Gibbs sampling method proposed in [116], the estimates $\hat{\theta}_i$ is sampled from the conditional distribution of θ_i, given all the other feature points and the observation vector $\mathbf{y}$. By assuming that $\{y_i\}_{i=1}^N$ are distributed according to the DPMM in (24), the conditional distribution of θ_i was obtained in [116] to be

Algorithm 1 Clustering algorithm.

Initialize $\hat{\theta}_i = y_i$, $\forall i \in \{1, \cdots, N\}$.
while Convergence condition not satisfied **do**
  for $i =$ shuffle $\{1, \cdots, N\}$ **do**
    Use Gibbs sampling to obtain $\hat{\theta}_i$ from the distribution in (25).
  end for
end while

$$\theta_i | \{\theta_j\}_{j \neq i}, \mathbf{y} = \left\{ \begin{array}{ll} \theta_j & \text{with Pr} = \frac{1}{B(y_i)} f_{\theta_j}(y_i) \\ \sim h(\theta | y_i) & \text{with Pr} = \frac{1}{B(y_i)} A(y_i) \end{array} \right. , \tag{25}$$

where $B(y_i) = A(y_i) + \sum_{l=1, l \neq i}^N f_{\theta_l}(y_i)$, $h(\theta_i | y_i) = \frac{\alpha_0}{A(y_i)} f_{\theta_i}(y_i) G_0(\theta_i)$ and $A(y) = \alpha_0 \int f_\theta(y) G_0(\theta) d\theta$.

In order to illustrate this clustering method, consider a simple example summarizing the process. We assume a set of mixture components $\theta \in \mathbb{R}$. Also, we assume $G_0(\theta)$ to be uniform over the range $[\theta_{min}, \theta_{max}]$. Note that this is a worst-case scenario assumption whenever there is no prior knowledge of the distribution of θ, except its range. Let $f_\theta(y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-\theta)^2}{2\sigma^2}}$.

Hence,

$$A(y) = \frac{\alpha_0}{\theta_{max} - \theta_{min}} \left[Q\left(\frac{\theta_{min} - y}{\sigma} \right) - Q\left(\frac{\theta_{max} - y}{\sigma} \right) \right] \tag{26}$$

and

$$h(\theta_i | y_i) = \left\{ \begin{array}{ll} B \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \theta_i)^2}{2\sigma^2}} & \text{if} \quad \theta_{min} \leq \theta_i \leq \theta_{max} \\ 0 & \text{otherwise} \end{array} \right. , \tag{27}$$

where $B = \frac{1}{Q\left(\frac{\theta_{min} - y_i}{\sigma}\right) - Q\left(\frac{\theta_{max} - y_i}{\sigma}\right)}$. Initially, we set $\theta_i = y_i$ for all $i \in \{1, \cdots, N\}$. The algorithm is described in Algorithm 1.

If the observation points $y_i \in \mathbb{R}^k$ (with $k > 1$), the distribution of $h(\theta_i | y_i)$ may become too complicated to be used in the sampling process of θ_i's. In [116], if $G_0(\theta)$ is constant in a large area around y_i, $h(\theta | y_i)$ was shown to be approximated by the Gaussian distribution (assuming that the observation pdf $f_\theta(y_i)$ is Gaussian). Thus, assuming a large uniform prior distribution on θ, we may approximate $h(\theta | y)$ by a Gaussian pdf so that (27) becomes:

$$h(\theta_i | y_i) = \mathcal{N}(y_i, \Sigma) \ , \tag{28}$$

where Σ is the covariance matrix.

In order to illustrate this approach in a multidimensional scenario, we may generate a Gaussian mixture model having 4 mixture components. The mixture components have different means in $\mathbb{R}^2$ and have an identity covariance matrix. We will assume that the covariance matrix is known.

We plot in Fig. 8 the results of the clustering algorithm based on DPMM. Three of the clusters were almost perfectly identified, whereas the forth cluster was split into three parts. The main advantage of this technique is its ability for learning the number of clusters from the data itself, without any prior knowledge. As opposed to heuristic or supervised classifi-

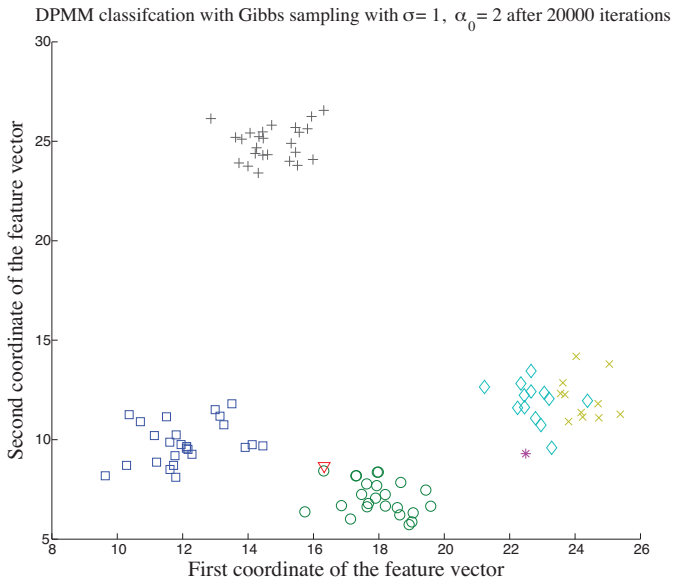


Fig. 8. The observation points y_i are classified into different clusters, denoted with different marker shapes. The original data points are generated from a Gaussian mixture model with 4 mixture components and with an identity covariance matrix.

cation approaches that assume a fixed number of clusters (such as the K-mean approach), the DPMM-based clustering technique is completely unsupervised, yet, provides effective classification results. This makes it a perfect choice for autonomous CRs that rely on unsupervised learning for decision-making, as suggested in [72].

4) Applications of Dirichlet process to cognitive radios: The Dirichlet process has been used as a framework for non-parametric Bayesian learning in CRs in [13], [148]. The approach was used for identifying and classifying wireless systems in [148], based on the CRP. The method consists of extracting two features from the observed signals (in particular, the center frequency and frequency spread) and to classify these feature points in a feature space by adopting an unsupervised clustering technique, based on the CRP. The objective is to identify both the number and types of wireless systems that exist in a certain frequency band at a certain moment. One application of this could be when multiple wireless systems co-exist in the same frequency band and try to communicate without interfering with each other. Such scenarios could arise in ISM bands where wireless local area networks (WLAN IEEE 802.11) coexist with wireless personal area networks (WPANs), such as Zigbee (IEEE 802.15.4) and Bluetooth (IEEE 802.15.1). In that case, a WPAN should sense the ISM band before selecting its communication channel so that it does not interfere with the WLAN or other WPAN systems. A realistic assumption in that case is that individual wireless users do not know the number of other coexisting wireless users. Instead, these unknown variables should be learnt based on appropriate autonomous learning algorithms. Moreover, the designed learning algorithms should account for the dynamics of the RF environment. For example, the number of wireless users might change over time. These dynamics should be handled by the embedded flexibility offered by non-parametric learning approaches.

The advantages of the Dirichlet process-based learning technique in [148] is that it does not rely on training data, making it suitable for identifying unknown signals via unsupervised learning. In this survey, we do not delve into details of choosing and computing appropriate feature points for the particular application considered in [148]. Instead, our focus below is on the implementation of the unsupervised learning and the associated clustering technique.

After sensing a certain signal, the CR extracts a feature point that captures certain spectrum characteristics. Usually, the extracted feature points are noisy and might be affected by estimation errors, receiver noise and path loss. Moreover, the statistical distribution of these observations might be unknown itself. It is expected that feature points that are extracted from a particular system will belong to the same cluster in the feature space. Depending on the feature definition, different systems might result in different clusters that are located at different places in the feature space. For example, if the feature point represents the center frequency, two systems transmitting at different carrier frequencies will result in feature points that are distributed around different mean points.

The authors in [148] argue that the clusters of a certain system are random themselves and might be drawn from a certain distribution. To illustrate this idea, assume two WiFi transmitters located at different distances from the receiver that both uses WLAN channel 1. Although the two transmitters belong to the same system (i.e. WiFi channel 1), their received powers might be different, resulting in variations of the features extracted from the signals of the same system. To capture this randomness, it can be assumed that the position and structure of the clusters formed (i.e. mean, variance, etc.) are themselves drawn from some distribution.

To be concrete, denote x as the derived feature point and assume that x is normally distributed with mean μ_c and covariance matrix Σ_c (i.e. $x \sim \mathcal{N}(\mu_c, \Sigma)$). These two parameters characterize a certain cluster and are drawn from a certain distribution. For example, it can be assumed that $\mu_c \sim \mathcal{N}(\mu_M, \Sigma_M)$ and $\Sigma_c \sim \mathcal{W}(V, n)$, where $\mathcal{W}$ denotes the Wishart distribution, which can be used to model the distribution of the covariance matrix of multivariate Gaussian variables.

In the method proposed in [148], a training stage[2] is required to estimate the parameters μ_M and Σ_M. This estimation can be performed by sensing a certain system (e.g. WiFi, or Zigbee) under different scenarios and estimating the centers of the clusters resulting from each experiment (i.e. estimating μ_c). The average of all μ_c's forms a maximum-likelihood (ML) estimate of the parameter μ_M of the corresponding wireless system. This step is equivalent to estimating the hyperparameters of a Dirichlet process [104]. A similar estimation method can also be performed to estimate Σ_M.

The knowledge of μ_M and Σ_M helps identify the corresponding wireless system of each cluster. That is, the maxi-

[2]Note that the training process used in [148] refers to the cluster formation process. The training used in [148] is done without data labeling nor human instructions, but with the CRP [144] and the Gibbs sampling [116], thus qualifying to be an unsupervised learning scheme.

mum a posteriori (MAP) detection can be applied to a cluster center μ_c to estimate the wireless system that it belongs to. However, the classification of feature points into clusters can be done based on the CRP.

The classification of a feature point into a certain cluster is made based on the Gibbs sampling applied to the CRP. The algorithm fixes the cluster assignments of all other feature points. Given that assignment, it generates a cluster index for the current feature point. This sampling process is applied to all the feature points separately until certain convergence criterion is satisfied. Other examples of the CRP-based feature classification can be found in speaker clustering [145] and document clustering applications [149].

B. Supervised Classification Methods in Cognitive Radios

Unlike the unsupervised learning techniques discussed in the previous section that may be used in alien environments without having any prior knowledge, supervised learning techniques can generally be used in familiar/known environments with prior knowledge about the characteristics of the environment. In the following, we introduce some of the major supervised learning techniques that have been applied to classification tasks in CRs.

1) Artificial Neural Network: The ANN has been motivated by the recognition that human brain computes in an entirely different way compared to the conventional digital computers [150]. A neural network is defined to be "a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use" [150]. An ANN resembles the brain in two respects [150]: 1) Knowledge is acquired by the network from its environment through a learning process and 2) interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.

Some of the most beneficial properties and capabilities of ANNs include: 1) Nonlinear fitness to underlying physical mechanisms, 2) adaptation ability to minor changes in surrounding environment and 3) providing information about the confidence in the decision made. However, the disadvantages of ANNs are that they require training under many different environment conditions and their training outcomes may depend crucially on the choice of initial parameters.

Various applications of ANNs to CRs can be found in recent literature [102], [151]–[155]. The authors in [151], for example, proposed the use of Multilayered Feedforward Neural Networks (MFNN) as a technique to synthesize performance evaluation functions in CRs. The benefit of using MFNNs is that they provide a general-purpose black-box modeling of the performance as a function of the measurements collected by the CR; furthermore, this characterization can be obtained and updated by a CR at run-time, thus effectively achieving a certain level of learning capability. The authors in [151] also demonstrated in several IEEE 802.11 based environments how these modeling capabilities can be used for optimizing the configuration of a CR.

In [152], the authors proposed an ANN-based cognitive engine that learns how environmental measurements and the status of the network affect its performance on different channels. In particular, an implementation of the proposed Cognitive Controller for dynamic channel selection in IEEE 802.11 wireless networks was presented. Performance evaluation carried out on an IEEE 802.11 wireless network deployment demonstrated that the Cognitive Controller is able to effectively learn how the network performance is affected by changes in the environment, and to perform dynamic channel selection thereby providing significant throughput enhancements.

In [153], an application of a Feedbackward ANN in conjunction with the cyclostationarity-based spectrum sensing was presented to perform spectrum sensing. The results showed that the proposed approach is able to detect the signals at considerably low signal-to-noise ratio (SNR) values. In [102], the authors designed a channel status predictor using a MFNN model. The authors argued that their proposed MFNN-based prediction is superior to the HMM based approaches, by pointing out that the HMM based approaches require a huge memory space to store a large number of past observations with high computational complexity.

In [154], the authors proposed a methodology for spectrum prediction by modeling licensed-user features as a multivariate chaotic time series, which is then input to an ANN that predicts the evolution of RF time series to decide if the unlicensed user can exploit the spectrum band. Experimental results showed a similar trend between predicted and observed values. This proposed spectrum evolution prediction method was done by exploiting the cyclostationary signal features to construct an RF multivariate time series that contain more information than the univariate time series, in contrast to most of the previously suggested modeling methodologies which focused on univariate time series prediction [156].

To illustrate the operation of ANNs in CR contexts, we present the model proposed in [78] and describe the main steps in the implementation of ANNs. In particular, [78] considers a multilayer perceptron (MLP) neural network which maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, which is fully connected from one layer to the next [78]. Except the input nodes, each node in the MLP is a neuron with a nonlinear activation function that computes a weighted sum of the up-layer output (denoted as the activation). An example of one of the most popular activation functions that is used in ANNs is the sigmoid function:

$$f(a) = \frac{1}{1 + e^{-a}} \ . \tag{29}$$

The ANN proposed in [78] has an input layer, output layer and multiple hidden layers. Note that, having additional hidden layers improves the nonlinear performance of the ANN in terms of classifying linearly non-separable data. However, adding more hidden layers makes the network more complicated and may require longer training time.

In the following, we consider an MLP network and let y_j^l to be the output of the j-th neuron in the l-th layer. Denote also by w_{ji}^l the weight between the j-th neuron in the l-th layer and the i-th neuron in the $l-1$-th layer. The output y_j^l

is given by:

$$y_j^l = \frac{1}{1 + e^{-\sum_i w_{ji}^l y_i^{l-1}}} \ . \tag{30}$$

During the training, the network tries to match the target value t_k to the output o_k of the k-th output neuron[3]. The error between the target and actual outputs is evaluated, for example, according to the mean-squared error (MSE):

$$MSE = \frac{1}{K} \sum_{k=1}^{K} (t_k - o_k)^2 \ , \tag{31}$$

where K is the number of output nodes. The update process will repeat until the MSE is smaller than a certain threshold.

The update rule can be performed according to a delta rule which adjusts the weights w_{ji}^l by an amount [78]:

$$\Delta w_{ji}^l = \eta \delta_j^l y_i^{l-1} \ , \tag{32}$$

where η is a learning rate and δ_j^l is defined as:

$$\delta_j^l = \begin{cases} o_j(t_j - o_j)(1 - o_j) & \text{if } l \text{ is the output layer} \\ y_j^l(1 - y_j^l) \sum_k \delta_k^{l+1} w_{kj}^{l+1} & \text{if } l \text{ is the hidden layer} \end{cases}$$

The authors in [78] used the above described MLP neural network to implement a learner in a cognitive engine. By assuming a WiMax configurable radio technology, the learner is able to choose a certain modulation mode according to the SNR, such that a certain bit-error rate (BER) will be achieved. Thus, the inputs of the neural network consists of the code rate and SNR values and the output is the resulting SNR. By supplying training data to the neural network, the cognitive engine is trained to identify the BER that results from a certain choice of modulation, given a certain SNR level. By comparing the performance of different scales of neural networks, the simulation results in [78] showed that increasing the number of hidden layers reduces the speed of convergence but leads to a smaller MSE. However, more training data are required for larger number of hidden layers. Thus, given a certain set of training data, a trade-off must be made between the speed of convergence and the convergence accuracy of the neural network.

2) Support Vector Machine: The SVM, developed by Vapnik and others [157], has been used for many machine learning tasks such as pattern recognition and object classifications. The SVM is characterized by the absence of local minima, the sparseness of the solution and the capacity control obtained by acting on the margin, or on other dimension independent quantities such as the number of support vectors [157]. SVM based techniques have achieved superior performances in a wide variety of real world problems due to their generalization ability and robustness against noise and outliers [158].

The basic idea of SVMs is to map the input vectors into a high-dimensional feature space in which they become linearly separable. This mapping from the input vector space to the feature space is a non-linear mapping which is achieved by using kernel functions. Depending on the application different types of kernel functions can be used. A common choice for classification problems is the Gaussian kernel which is a polynomial kernel of infinite degree. In performing classification, a hyperplane which allows for the largest generalization in this high-dimensional space is found. This is so-called a maximal margin classifier [159]. Note that, the margin is defined as the distance from a separating hyperplane to the closest data points. As shown in Fig. 9, there could be many possible separating hyperplanes between the two classes of data, but only one of them allows for the maximum margin. The corresponding closest data points are named support vectors and the hyperplane allowing for the maximum margin is called an optimal separating hyperplane. The interested reader is referred to [79], [160], [161] for insightful discussion on SVMs.

[3]Since a certain target value (i.e. a label) is required during the training process, neural networks are considered as supervised learning algorithms.

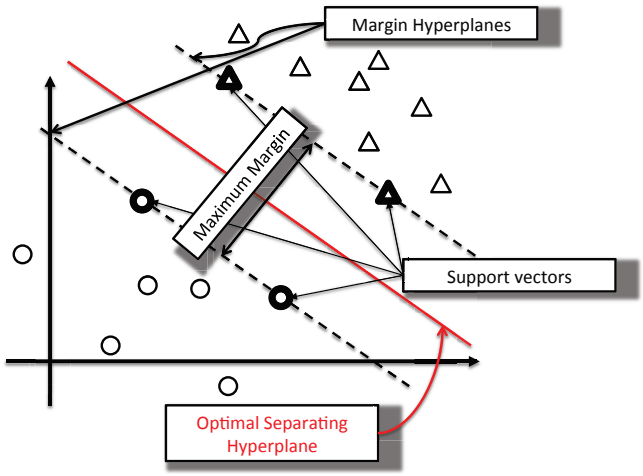


Fig. 9. A diagram showing the basic idea of SVM: optimal separation hyperplane (solid red line) and two margin hyperplanes (dashed lines) in a binary classification example; Support vectors are bolded.

An SVM-based classifier was described in [161] for signal classification in CRs. The classifier in [161] assumed a training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^l$ with $x \in \mathbb{R}^N$ and $y \in \{-1, 1\}$. The objective is to find a hyperplane:

$$\mathbf{w}^T \varphi(\mathbf{x}) + b = 0 \ , \tag{33}$$

where φ can be a non-linear function that maps $\mathbf{x}$ into a higher dimensional Hilbert space [160], $\mathbf{w}$ is a weight vector and b is a scalar parameter. In general, it is not possible to obtain an expression for the mapping function φ. However, this function can be characterized by a Kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ and, as it turns out fortunately, the Kernel function is sufficient to optimize the parameters $\mathbf{w}$ and b in (33) [160].

The hyperplane in (33) is assumed to separate the data into two classes such that the distance between the closest points of each class to the hyperplane is maximized. This can be achieved by minimizing the norm $\|\mathbf{w}\|^2$ [160].

In order to solve the optimization problem, the slacks variables $\{\xi_i, i = 1, \cdots, l\}$ are introduced and the optimization problem can be formulated as [161]:

$$\min_{\mathbf{w}, b, \xi_i} \ \tfrac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^l \xi_i \tag{34}$$

$$\text{s.t. } y_i \left(\mathbf{w}^T \varphi(\mathbf{x}_i) + b \right) \geq 1 - \xi_i, \forall i = 1, \cdots, l \tag{35}$$

$$\xi_i \geq 0, \forall i = 1, \cdots, l \tag{36}$$

where C is the penalty parameter that controls the training error.

The Lagrangian of the above optimization problem can be written as:

$$L = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\xi_i - \sum_{i=1}^{l}\beta_i\xi_i - \\ - \sum_{i=1}^{l}\alpha_i\left[\mathbf{w}^T\varphi(\mathbf{x}_i+b) - 1 + \xi_i\right] ,$$

where $\alpha_i, \beta_i \geq 0$ are the Lagrange multipliers. By computing the derivatives with respect to $\mathbf{w}$, b and ξ_i, the dual representation of the optimization problem can be expressed as [161]:

$$\max_{(\alpha_1,\cdots,\alpha_l)} \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j) \\ \text{s.t. } 0 \leq \alpha_i \leq C, \forall i = 1,\cdots,l \\ \sum_{i=1}^{l} y_i\alpha_i = 0$$

where $K(\mathbf{x}_i,\mathbf{x}_j) = \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$ is the Kernel function.

In this case, the decision function (i.e. the learning machine [160]) is computed as:

$$f(x) = sgn\left\{\sum_{i=1}^{l}\alpha_i y_i K(\mathbf{x}_i,\mathbf{x}) + b\right\} . \quad (37)$$

Other applications of SVMs to CR can be found in current literature, including [65], [79], [103], [158], [161]–[167]. Most of these applications of the SVM in CR context, however, has been for performing signal classification.

In [164], for example, a MAC protocol classification scheme was proposed to classify contention-based and control-based MAC protocols in an unknown primary network based on SVMs. To perform the classification in an unknown primary network, the mean and variance of the received power are chosen as two features for the SVM. The SVM is embedded in a CR terminal of the secondary network. A TDMA and a slotted Aloha network were setup as the primary networks. Simulation results showed that TDMA and slotted Aloha MAC protocol could be effectively classified by the CR terminal and the correct classification rate was proportional to the transmission rate of the primary networks, where the transmission rate for the primary networks is defined as the new packet generating/arriving probability in each time slot. The reason for the increase in the correct classification rate when the transmission rate increases is the following: for slotted Aloha network, the higher transmission rate brings the higher collision probability, and thus the higher instantaneous received power captured by a CR terminal; for TDMA network, however, there is no relation between transmission rate and instantaneous captured received power. Therefore, when the transmission rates of both primary networks increase, it makes a CR terminal easier to differentiate TDMA and slotted Aloha.

SVM classifiers can not only be a binary classifier as shown in the previous example, but also it can be easily used as a multi-class classifiers by treating a K-class classification problem as K two-class problems. For example, in [165] the authors presented a study of multi-class signal classification based on automatic modulation classification (AMC) through SVMs. A simulated model of an SVM signal classifier was implemented and trained to recognize seven distinct modulation schemes; five digital (BPSK, QPSK, GMSK, 16-QAM and 64-QAM) and two analog (FM and AM). The signals were generated using realistic carrier frequency, sampling frequency and symbol rate values, and realistic Raised- cosine and Gaussian pulse-shaping filters. The results showed that the implemented classifier can correctly classify signals with high probabilities.

V. Centralized and Decentralized Learning in Cognitive Radio

Since noise uncertainties, shadowing, and multi-path fading effects limit the performance of spectrum sensing, when the received primary SNR is too low, there exists a SNR wall, below which reliable spectrum detection is impossible in some cases [168], [169]. If secondary users cannot detect the primary transmitter, while the primary receiver is within the secondary users transmission range, a hidden terminal problem occurs [170], [171], and the primary user's transmission will be interfered with. By taking advantage of diversity offered by multiple independent fading channels (multiuser diversity), cooperative spectrum sensing improves the reliability of spectrum sensing and the utilization of idle spectrum [25], [26], as opposed to non-cooperative spectrum sensing.

In centralized cooperative spectrum sensing [25], [26], a central controller collects local observations from multiple secondary users, decides the spectrum occupancy by using decision fusion rules, and informs the secondary users which channels to access. In distributed cooperative spectrum sensing [55], [172], on the other hand, secondary users within a CRN exchange their local sensing results among themselves without requiring a backbone or centralized infrastructure. On the other hand, in the non-cooperative decentralized sensing framework, no communications are assumed among the secondary users [173].

In [174], the authors showed how various centralized and decentralized spectrum access markets (where CRs can compete over time for dynamically available transmission opportunities) can be designed based on a stochastic game (discussed above in Section III-C) framework and solved using a learning algorithm. Their proposed learning algorithm was to learn the following information in the stochastic game: state transition model, state and the policy of other secondary users and the network resource state. The proposed learning algorithm was similar to Q-learning. However, the main difference compared to Q-learning was that it explicitly considered the impact of other secondary user actions through the state classifications and transition probability approximation. The computational complexity and performance were also discussed in [174].

In [37] the authors proposed and analyzed both a centralized and a decentralized decision-making architecture with RL for the secondary CRN. In this work, a new way to encourage primary users to lease their spectrum was proposed: the secondary users place bids indicating how much power they are willing to spend for relaying the primary signals to their destinations. In this formulation, the primary users achieve power savings due to asymmetric cooperation. In the

		Spectrum Sensing and MAC Protocols	Signal Classification and Feature Detection	Power Allocation and Rate adaptation	System Parameters Reconfiguration	Pros	Cons
Unsupervised learning techniques	Reinforcement learning (RL)	x				Optimal solution for MDP's	In general, suboptimal for POMDP's, DEC-MDP's and DEC-POMDP's
	Non-parametric Learning: DPMM		x			Does not require prior knowledge about the number of mixture components	Requires large number of iterations, compared to parametric methods
	Game theory-based Learning	x		x		Suitable for multi-player decision problems	Requires knowledge of different parameters (e.g. SINR, power, price from base stations, etc.) which is impractical in many situations
	Threshold Learning				x	Suitable for controlling specific parameters under uncertainty conditions	Requires training data
Supervised learning techniques	Artificial Neural Network (ANN)		x			Does not require prior knowledge of the distribution of the observed process	▪ Suffers from *overfitting* ▪ Requires data labeling
	Support Vector Machine (SVM)		x			Has better performance for small training examples, compared to ANN	▪ Requires prior knowledge of the distribution of the observed process ▪ Requires data labeling

Fig. 10. A comparison among the learning algorithms that are presented in this survey.

centralized architecture, a secondary system decision center (SSDC) selects a bid for each primary channel based on optimal channel assignment for secondary users. In a decentralized CRN architecture, an auction game-based protocol was proposed in which each secondary user independently places bids for each primary channel and receivers of each primary link pick the bid that will lead to the most power savings. A simple and robust distributed RL mechanism was developed to allow the users to revise their bids and to increase their subsequent rewards. The performance results given in [37] showed the significant impact of RL in both improving spectrum utilization and meeting individual secondary user performance requirements.

In [12], the authors considered DSA among CRs from an adaptive, game theoretic learning perspective, in which CRs compete for channels temporarily vacated by licensed primary users in order to satisfy their own demands while minimizing interference. For both slowly varying primary user activity and slowly varying statistics of fast primary user activity, the authors applied an adaptive regret based learning procedure which tracks the set of correlated equilibria of the game, treated as a distributed stochastic approximation. The proposed approach was decentralized in terms of both radio awareness and activity; radios estimate spectral conditions based on their own experience, and adapt by choosing spectral allocations which yield them the greatest utility. Iterated over time, this process converges so that each radio's performance is an optimal response to others' activity. This apparently selfish scheme was also used to deliver system-wide performance by a judicious choice of utility function. This procedure was shown to perform well compared to other similar adaptive algorithms. The results of the estimation of channel contention for a simple carrier sense multiple access (CSMA) channel sharing scheme was also presented.

In [175], the authors proposed an auction framework for CRNs to allow secondary users to share the available spectrum of licensed primary users fairly and efficiently, subject to the interference temperature constraint at each primary user. The competition among secondary users was studied by formulating a non-cooperative multiple-primary users multiple-secondary users auction game. The resulting equilibrium was found by solving a non-continuous two-dimensional optimization problem. A distributed algorithm was also developed in which each secondary user updates its strategy based on local information to converge to the equilibrium. The proposed auction framework was then extended to the more challenging scenario with free spectrum bands. An algorithm was developed based on the no-regret learning to reach a correlated equilibrium of the auction game. The proposed algorithm, which can be implemented distributedly based on local observation, is especially suited in decentralized adaptive learning environments. The authors demonstrated the effectiveness of the proposed auction framework in achieving high efficiency and fairness in spectrum allocation through numerical examples.

In general, there is always a trade-off between the centralized and decentralized control in radio networks. This is also true for CRNs. While the centralized schemes ensure efficient management of the spectrum resources, they often suffer from signaling and processing overhead. On the other hand, a decentralized scheme can reduce the complexity of the decision-making in cognitive networks. However, radios

that act according to a decentralized scheme may adopt a selfish behavior and try to maximize their own utilities, at the expense of the sum-utility of the network (social welfare), leading to overall network inefficiency. This problem can become particularly severe when considering heterogeneous networks in which different nodes belong to different types of systems and have different objectives (usually conflicting objectives). To resolve this problem, [176] proposes a hybrid approach for heterogeneous CRNs where the wireless users are assisted in their decisions by the network which broadcasts aggregated information to the users [176]. At some states of the system, the network manager imposes its decisions on users in the network. In other states, the mobile nodes may take autonomous actions in response to the information sent by the network center. As a result, the model in [176] avoids having a completely decentralized network, due to possible inefficiency of such non-cooperative networks. Nevertheless, a large part of the decision-making is still delegated to the mobile nodes to reduce the processing overhead at the central node.

In the problem formulation of [176], the authors consider a wireless network composed of S systems that are managed by the same operator. The set of all serving systems is denoted by $\mathcal{S} = \{1, \cdots, S\}$. Since the throughput of each serving system drops as a function of the distance of between the mobile and the base station, the throughput of a mobile changes within a given cell. To capture this variation, each cell is split into N circles of radius d_n ($n \in \mathcal{N} = \{1, \cdots, N\}$). Each circle area is assumed to have the same radio characteristics. In this case, all mobile systems that are located within circle $n \in \mathcal{N}$ and are served by system $s \in \mathcal{S}$ achieve the same throughput. The network state matrix is denoted by $\mathbf{M} \in \mathcal{F}$, where $\mathcal{F} = \mathbb{N}^{N \times S}$. The (n, s)-th element M_n^s of the matrix $\mathbf{M}$ denotes the number of users with radio condition $n \in \mathcal{N}$ which are served by system $s \in \mathcal{S}$ in the circle. The network is fully characterized by its state $\mathbf{M}$, but this information is not available to the mobile nodes when the radio resource management (RRM) is decentralized. In this case, by using the *radio enabler* proposed in IEEE 1900.4, the network reconfiguration manager (NRM) broadcasts to the terminal reconfiguration manager (TRM) an aggregated load information that takes values in some finite set $\mathcal{L} = \{1, \cdots, L\}$ indicating whether the load state at mobile terminals are either low, medium or high. The mapping $f : \mathbf{M} \mapsto \mathcal{L}$ specifies a macro-state $f(\mathbf{M})$ for each network micro-state $\mathbf{M}$. This state encoding reduces the signaling overhead, while satisfying the requirements of the IEEE 1900.4 standard which state that "the network manager side shall periodically update the terminal side with context information" [177]. Given the load information $l = f(\mathbf{M})$ and the radio condition $n \in \mathcal{N}$, the mobile makes its decision $P_{n,l} \in \mathcal{S}$, specifying which system it will connect to, and the user's decision vector is denoted by $\mathbf{P}^l = [P_{1,l} \cdots, P_{N,l}]$.

The authors in [176] find the association policies by following three different approaches:

1) Global optimum approach.
2) Nash equilibrium approach.
3) Stackelberg game approach.

The global optimum approach finds the policy that maximizes the global utility of the network. However, since it is not realistic to consider that individual users will seek the global optimum, another policy (corresponding to the Nash equilibrium) was obtained such that it maximizes the users' utilities. Finally, a Stackelberg game formulation was developed for the operator to control the equilibrium of its wireless users. This leads to maximizing the operator's utility by sending appropriate load information $l \in \mathcal{L}$ to the distributed radios.

The authors of [176] analyzed the network performance under these three different association policies. They demonstrated, by means of Stackelberg formulation, how the operator can optimize its global utility by sending appropriate information about the network state, while users maximize their individual utilities. The resulting hybrid architecture achieved a good trade-off between the global network performance and the signaling overhead, making it a viable alternative to be considered when designing CRNs.

VI. Conclusion

In this survey paper, we have characterized the learning problems in CRs and stated the importance of machine learning in developing real CRs. We have presented the state-of-the-art learning methods that have been applied to CRs classifying them under supervised and unsupervised learning. A discussion of some of the most important, and commonly used, learning algorithms was provided along with their advantages and disadvantages. We also showed some of the challenging learning problems encountered in CRs and presented possible solution methods to address them.

References

[1] J. Mitola III and G. Q. Maguire, Jr., "Cognitive radio: making software radios more personal," *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 13 –18, Aug. 1999.

[2] J. Mitola, "Cognitive radio: An integrated agent architecture for software defined radio," Ph.D. dissertation, Royal Institute of Technology (KTH), Stockholm, Sweden, 2000.

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

[4] T. Costlow, "Cognitive radios will adapt to users," *IEEE Intell. Syst.*, vol. 18, no. 3, p. 7, May-June 2003.

[5] S. K. Jayaweera and C. G. Christodoulou, "Radiobots: Architecture, algorithms and realtime reconfigurable antenna designs for autonomous, self-learning future cognitive radios," University of New Mexico, Technical Report EECE-TR-11-0001, Mar. 2011. [Online]. Available: http://repository.unm.edu/handle/1928/12306

[6] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[7] FCC, "Report of the spectrum efficiency working group," FCC spectrum policy task force, Tech. Rep., Nov. 2002.

[8] ——, "ET docket no 03-322 notice of proposed rulemaking and order," Tech. Rep., Dec. 2003.

[9] N. Devroye, M. Vu, and V. Tarokh, "Cognitive radio networks," *IEEE Signal Processing Mag.*, vol. 25, pp. 12–23, Nov. 2008.

[10] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa, "Breaking spectrum gridlock with cogntive radios: An information theoretic perspective," *Proc. IEEE*, vol. 97, no. 5, pp. 894–914, May 2009.

[11] V. Krishnamurthy, "Decentralized spectrum access amongst cognitive radios - An interacting multivariate global game-theoretic approach," *IEEE Trans. Signal Process.*, vol. 57, no. 10, pp. 3999 –4013, Oct. 2009.

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

[13] Z. Han, R. Zheng, and H. Poor, "Repeated auctions with Bayesian nonparametric learning for spectrum access in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 3, pp. 890 –900, Mar. 2011.

[14] J. Lunden, V. Koivunen, S. Kulkarni, and H. Poor, "Reinforcement learning based distributed multiagent sensing policy for cognitive radio networks," in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '11)*, Aachen, Germany, May 2011, pp. 642 –646.

[15] K. Ben Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proc. IEEE*, vol. 97, no. 5, pp. 878 –893, May 2009.

[16] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Processing Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.

[17] S. K. Jayaweera and T. Li, "Dynamic spectrum leasing in cognitive radio networks via primary-secondary user power control games," *IEEE Trans. Wireless Commun.*, vol. 8, no. 6, pp. 3300–3310, July 2009.

[18] S. K. Jayaweera, , G. Vazquez-Vilar, and C. Mosquera, "Dynamic spectrum leasing: A new paradigm for spectrum sharing in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 5, pp. 2328–2339, May 2010.

[19] G. Zhao, J. Ma, Y. Li, T. Wu, Y. H. Kwon, A. Soong, and C. Yang, "Spatial spectrum holes for cognitive radio with directional transmission," in *IEEE Global Telecommunications Conference (GLOBECOM '08)*, Nov. 2008, pp. 1–5.

[20] A. Ghasemi and E. Sousa, "Spectrum sensing in cognitive radio networks: requirements, challenges and design trade-offs," *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 32–39, Apr. 2008.

[21] B. Farhang-Boroujeny, "Filter bank spectrum sensing for cognitive radios," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1801–1811, May 2008.

[22] B. Farhang-Boroujeny and R. Kempter, "Multicarrier communication techniques for spectrum sensing and communication in cognitive radios," *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 80–85, Apr. 2008.

[23] C. R. C. da Silva, C. Brian, and K. Kyouwoong, "Distributed spectrum sensing for cognitive radio systems," in *Information Theory and Applications Workshop*, Feb. 2007, pp. 120–123.

[24] Y. Li, S. Jayaweera, M. Bkassiny, and K. Avery, "Optimal myopic sensing and dynamic spectrum access in cognitive radio networks with low-complexity implementations," *IEEE Trans. Wireless Commun.*, vol. 11, no. 7, pp. 2412 –2423, July 2012.

[25] ——, "Optimal myopic sensing and dynamic spectrum access in centralized secondary cognitive radio networks with low-complexity implementations," in *IEEE 73rd Vehicular Technology Conference (VTC-Spring '11)*, May 2011, pp. 1 –5.

[26] M. Bkassiny, S. K. Jayaweera, Y. Li, and K. A. Avery, "Optimal and low-complexity algorithms for dynamic spectrum access in centralized cognitive radio networks with fading channels," in *IEEE Vehicular Technology Conference (VTC-spring '11)*, Budapest, Hungary, May 2011.

[27] C. Cordeiro, M. Ghosh, D. Cavalcanti, and K. Challapali, "Spectrum sensing for dynamic spectrum access of TV bands," in *2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom '07)*, Aug. 2007, pp. 225–233.

[28] H. Chen, W. Gao, and D. G. Daut, "Signature based spectrum sensing algorithms for IEEE 802.22 WRAN," in *IEEE International Conference on Communications (ICC '07)*, June 2007, pp. 6487–6492.

[29] Y. Zeng and Y. Liang, "Maximum-minimum eigenvalue detection for cognitive radio," in *18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '07)*, Sep. 2007, pp. 1–5.

[30] ——, "Covariance based signal detections for cognitive radio," in *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '07)*, Apr. 2007, pp. 202–207.

[31] X. Zhou, Y. Li, Y. H. Kwon, and A. Soong, "Detection timing and channel selection for periodic spectrum sensing in cognitive radio," in *IEEE Global Telecommunications Conference (GLOBECOM '08)*, Nov. 2008, pp. 1–5.

[32] Z. Tian and G. B. Giannakis, "A wavelet approach to wideband spectrum sensing for cognitive radios," in *1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, June 2006, pp. 1–5.

[33] G. Ganesan and Y. Li, "Cooperative spectrum sensing in cognitive radio, part I: Two user networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 6, pp. 2204–2213, June 2007.

[34] ——, "Cooperative spectrum sensing in cognitive radio, part II: Multiuser networks," *Wireless Communications, IEEE Trans.on*, vol. 6, no. 6, pp. 2214–2222, June 2007.

[35] Y. Chen, Q. Zhao, and A. Swami, "Joint design and separation principle for opportunistic spectrum access in the presence of sensing errors," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 2053 –2071, May 2008.

[36] S. Huang, X. Liu, and Z. Ding, "Opportunistic spectrum access in cognitive radio networks," in *The 27th Conference on Computer Communications (IEEE INFOCOM '08)*, Phoenix, AZ, Apr. 2008, pp. 1427 –1435.

[37] S. Jayaweera, M. Bkassiny, and K. Avery, "Asymmetric cooperative communications based spectrum leasing via auctions in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 8, pp. 2716 –2724, Aug. 2011.

[38] M. Bkassiny, S. K. Jayaweera, and K. A. Avery, "Distributed reinforcement learning based MAC protocols for autonomous cognitive secondary users," in *20th Annual Wireless and Optical Communications Conference (WOCC '11)*, Newark, NJ, Apr. 2011, pp. 1 –6.

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

[40] S. Haykin, D. Thomson, and J. Reed, "Spectrum sensing for cognitive radio," *Proc. IEEE*, vol. 97, no. 5, pp. 849 –877, May 2009.

[41] J. Ma, G. Y. Li, and B. H. Juang, "Signal processing in cognitive radio," *Proc. IEEE*, vol. 97, no. 5, pp. 805–823, May 2009.

[42] W. Zhang, R. Mallik, and K. Letaief, "Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 12, pp. 5761 –5766, Dec. 2009.

[43] Y. M. Kim, G. Zheng, S. H. Sohn, and J. M. Kim, "An alternative energy detection using sliding window for cognitive radio system," in *10th International Conference on Advanced Communication Technology (ICACT '08)*, vol. 1, Gangwon-Do, South Korea, Feb. 2008, pp. 481 –485.

[44] J. Lunden, V. Koivunen, A. Huttunen, and H. Poor, "Collaborative cyclostationary spectrum sensing for cognitive radio systems," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4182 –4195, Nov. 2009.

[45] A. Dandawate and G. Giannakis, "Statistical tests for presence of cyclostationarity," *IEEE Trans. Signal Process.*, vol. 42, no. 9, pp. 2355 –2369, Sep. 1994.

[46] B. Deepa, A. Iyer, and C. Murthy, "Cyclostationary-based architectures for spectrum sensing in IEEE 802.22 WRAN," in *IEEE Global Telecommunications Conference (GLOBECOM '10)*, Miami, FL, Dec. 2010, pp. 1 –5.

[47] M. Gandetto and C. Regazzoni, "Spectrum sensing: A distributed approach for cognitive terminals," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 546 –557, Apr. 2007.

[48] J. Unnikrishnan and V. Veeravalli, "Cooperative sensing for primary detection in cognitive radio," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 18 –27, Feb. 2008.

[49] T. Cui, F. Gao, and A. Nallanathan, "Optimization of cooperative spectrum sensing in cognitive radio," *IEEE Trans. Veh. Technol.*, vol. 60, no. 4, pp. 1578 –1589, May 2011.

[50] O. Simeone, I. Stanojev, S. Savazzi, Y. Bar-Ness, U. Spagnolini, and R. Pickholtz, "Spectrum leasing to cooperating secondary ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 26, pp. 203–213, Jan. 2008.

[51] Q. Zhang, J. Jia, and J. Zhang, "Cooperative relay to improve diversity in cognitive radio networks," *IEEE Commun. Mag.*, vol. 47, no. 2, pp. 111 –117, Feb. 2009.

[52] Y. Han, A. Pandharipande, and S. Ting, "Cooperative decode-and-forward relaying for secondary spectrum access," *IEEE Trans. Wireless Commun.*, vol. 8, no. 10, pp. 4945 –4950, Oct. 2009.

[53] L. Li, X. Zhou, H. Xu, G. Li, D. Wang, and A. Soong, "Simplified relay selection and power allocation in cooperative cognitive radio systems," *IEEE Trans. Wireless Commun.*, vol. 10, no. 1, pp. 33 –36, Jan. 2011.

[54] E. Hossain and V. K. Bhargava, *Cognitive Wireless Communication Networks*. Springer, 2007.

[55] B. Wang and K. J. R. Liu, "Advances in cognitive radio networks: A survey," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 5 –23, Feb. 2011.

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

[57] K. Shin, H. Kim, A. Min, and A. Kumar, "Cognitive radios for dynamic spectrum access: from concept to reality," *IEEE Wireless Commun.*, vol. 17, no. 6, pp. 64 –74, Dec. 2010.

[58] A. De Domenico, E. Strinati, and M.-G. Di Benedetto, "A survey on MAC strategies for cognitive radio networks," *IEEE Commun. Surveys Tutorials*, vol. 14, no. 1, pp. 21 –44, quarter 2012.

[59] A. Mody, M. Sherman, R. Martinez, R. Reddy, and T. Kiernan, "Survey of IEEE standards supporting cognitive radio and dynamic spectrum access," in *IEEE Military Communications Conference (MILCOM '08)*, Nov. 2008, pp. 1 –7.

[60] Q. Zhao and A. Swami, "A survey of dynamic spectrum access: Signal processing and networking perspectives," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '07)*, vol. 4, Apr. 2007, pp. IV–1349 –IV–1352.

[61] J. Mitola, "Cognitive radio architecture evolution," *Proc. IEEE*, vol. 97, no. 4, pp. 626 –641, Apr. 2009.

[62] S. Jayaweera, Y. Li, M. Bkassiny, C. Christodoulou, and K. Avery, "Radiobots: The autonomous, self-learning future cognitive radios," in *International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS '11)*, Chiangmai, Thailand, Dec. 2011, pp. 1 –5.

[63] A. El-Saleh, M. Ismail, M. Ali, and J. Ng, "Development of a cognitive radio decision engine using multi-objective hybrid genetic algorithm," in *IEEE 9th Malaysia International Conference on Communications (MICC 2009)*, Dec. 2009, pp. 343 –347.

[64] L. Morales-Tirado, J. Suris-Pietri, and J. Reed, "A hybrid cognitive engine for improving coverage in 3G wireless networks," in *IEEE International Conference on Communications Workshops (ICC Workshops 2009).*, June 2009, pp. 1 –5.

[65] Y. Huang, H. Jiang, H. Hu, and Y. Yao, "Design of learning engine based on support vector machine in cognitive radio," in *International Conference on Computational Intelligence and Software Engineering (CiSE '09)*, Wuhan, China, Dec. 2009, pp. 1 –4.

[66] Y. Huang, J. Wang, and H. Jiang, "Modeling of learning inference and decision-making engine in cognitive radio," in *Second International Conference on Networks Security Wireless Communications and Trusted Computing (NSWCTC)*, vol. 2, Apr. 2010, pp. 258 –261.

[67] Y. Yang, H. Jiang, and J. Ma, "Design of optimal engine for cognitive radio parameters based on the DUGA," in *3rd International Conference on Information Sciences and Interaction Sciences (ICIS 2010)*, June 2010, pp. 694 –698.

[68] H. Volos and R. Buehrer, "Cognitive engine design for link adaptation: An application to multi-antenna systems," *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2902 –2913, Sep. 2010.

[69] C. Clancy, J. Hecker, E. Stuntebeck, and T. O'Shea, "Applications of machine learning to cognitive radio networks," *IEEE Wireless Commun.*, vol. 14, no. 4, pp. 47 –52, Aug. 2007.

[70] A. N. Mody, S. R. Blatt, N. B. Thammakhoune, T. P. McElwain, J. D. Niedzwiecki, D. G. Mills, M. J. Sherman, and C. S. Myers, "Machine learning based cognitive communications in white as well as the gray space," in *IEEE Military Communications Conference. (MILCOM '07)*, Orlando, FL, Oct. 2007, pp. 1 –7.

[71] M. Bkassiny, S. K. Jayaweera, Y. Li, and K. A. Avery, "Wideband spectrum sensing and non-parametric signal classification for autonomous self-learning cognitive radios," *IEEE Trans. Wireless Commun.*, vol. 11, no. 7, pp. 2596 –2605, July 2012.

[72] ——, "Blind cyclostationary feature detection based spectrum sensing for autonomous self-learning cognitive radios," in *IEEE International Conference on Communications (ICC '12)*, Ottawa, Canada, June 2012.

[73] X. Gao, B. Jiang, X. You, Z. Pan, Y. Xue, and E. Schulz, "Efficient channel estimation for MIMO single-carrier block transmission with dual cyclic timeslot structure," *IEEE Trans. Commun.*, vol. 55, no. 11, pp. 2210 –2223, Nov. 2007.

[74] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[75] S. Gong, W. Liu, W. Yuan, W. Cheng, and S. Wang, "Threshold-learning in local spectrum sensing of cognitive radio," in *IEEE 69th Vehicular Technology Conference (VTC Sp. '09)*, Barcelona, Spain, Apr. 2009, pp. 1 –6.

[76] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: John Wiley and Sons, 1994.

[77] A. Galindo-Serrano and L. Giupponi, "Distributed Q-learning for aggregated interference control in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1823 –1834, May 2010.

[78] X. Dong, Y. Li, C. Wu, and Y. Cai, "A learner based on neural network for cognitive radio," in *12th IEEE International Conference on Communication Technology (ICCT '10)*, Nanjing, China, Nov. 2010, pp. 893 –896.

[79] M. M. Ramon, T. Atwood, S. Barbin, and C. G. Christodoulou, "Signal classification with an SVM-FFT approach for feature extraction in cognitive radio," in *SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC '09)*, Belem, Brazil, Nov. 2009, pp. 286 –289.

[80] B. Hamdaoui, P. Venkatraman, and M. Guizani, "Opportunistic exploitation of bandwidth resources through reinforcement learning," in *IEEE Global Telecommunications Conference (GLOBECOM '09)*, Honolulu, HI, Dec. 2009, pp. 1 –6.

[81] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal, "Applications of reinforcement learning to cognitive radio networks," in *IEEE International Conference on Communications Workshops (ICC), 2010*, Cape Town, South Africa, May 2010, pp. 1 –6.

[82] Y. Reddy, "Detecting primary signals for efficient utilization of spectrum using Q-learning," in *Fifth International Conference on Information Technology: New Generations (ITNG '08)*, Las Vegas, NV, Apr. 2008, pp. 360 –365.

[83] M. Li, Y. Xu, and J. Hu, "A Q-learning based sensing task selection scheme for cognitive radio networks," in *International Conference on Wireless Communications Signal Processing (WCSP '09)*, Nanjing, China, Nov. 2009, pp. 1 –5.

[84] Y. Yao and Z. Feng, "Centralized channel and power allocation for cognitive radio networks: A Q-learning solution," in *Future Network and Mobile Summit*, Florence, Italy, June 2010, pp. 1 –8.

[85] P. Venkatraman, B. Hamdaoui, and M. Guizani, "Opportunistic bandwidth sharing through reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 59, no. 6, pp. 3148 –3153, July 2010.

[86] T. Jiang, D. Grace, and P. Mitchell, "Efficient exploration in reinforcement learning-based cognitive radio spectrum sharing," *IET Communications*, vol. 5, no. 10, pp. 1309 –1317, Jan. 2011.

[87] T. Clancy, A. Khawar, and T. Newman, "Robust signal classification using unsupervised learning," *IEEE Trans. Wireless Commun.*, vol. 10, no. 4, pp. 1289 –1299, Apr. 2011.

[88] C. Claus and C. Boutilier, "The dynamics of reinforcement learning in cooperative multiagent systems," in *Proc. Fifteenth National Conference on Artificial Intelligence*, Madison, WI, Jul. 1998, pp. 746–752.

[89] G. D. Croon, M. F. V. Dartel, and E. O. Postma, "Evolutionary learning outperforms reinforcement learning on non-Markovian tasks," in *8th European Conference on Artificial Life Workshop on Memory and Learning Mechanisms in Autonomous Robots*, Canterbury, Kent, UK, 2005.

[90] R. Sutton, D. Mcallester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. 12th conference on Advances in Neural Information Processing Systems (NIPS '99)*. Denver, CO: MIT Press, 2001, pp. 1057–1063.

[91] J. Baxter and P. L. Bartlett, "Infinite-horizon policy-gradient estimation," *Journal of Artificial Intelligence Research*, vol. 15, pp. 319–350, 2001.

[92] D. E. Moriarty, A. C. Schultz, and J. J. Grefenstette, "Evolutionary algorithms for reinforcement learning," *J. Artificial Intelligence Research*, vol. 11, pp. 241–276, 1999.

[93] F. Dandurand and T. Shultz, "Connectionist models of reinforcement, imitation, and instruction in learning to solve complex problems," *IEEE Trans. Autonomous Mental Development*, vol. 1, no. 2, pp. 110 –121, Aug. 2009.

[94] Y. Xing and R. Chandramouli, "Human behavior inspired cognitive radio network design," *IEEE Commun. Mag.*, vol. 46, no. 12, pp. 122 –127, Dec. 2008.

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

[96] B. Wang, K. Ray Liu, and T. Clancy, "Evolutionary cooperative spectrum sensing game: how to collaborate?" *IEEE Trans. Commun.*, vol. 58, no. 3, pp. 890 –900, Mar. 2010.

[97] A. Galindo-Serrano, L. Giupponi, P. Blasco, and M. Dohler, "Learning from experts in cognitive radio networks: The docitive paradigm," in *Proc. Fifth International Conference on Cognitive Radio Oriented Wireless Networks Communications (CROWNCOM '10)*, Cannes, France, June 2010, pp. 1 –6.

[98] 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 Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1578 –1592, May 2010.

[99] R. S. Michalski, "Learning and cognition," in *World Conference on the Fundamentals of Artificial Intelligence (WOCFAI '95)*, Paris, France, July 1995, pp. 507–510.

[100] J. Burbank, A. Hammons, and S. Jones, "A common lexicon and design issues surrounding cognitive radio networks operating in the presence of jamming," in *IEEE Military Communications Conference (MILCOM '08)*, San Diego, CA, Nov. 2008, pp. 1 –7.

[101] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[102] V. Tumuluru, P. Wang, and D. Niyato, "A neural network based spectrum prediction scheme for cognitive radio," in *IEEE International Conference on Communications (ICC '10)*, May 2010, pp. 1 –5.

[103] H. Hu, J. Song, and Y. Wang, "Signal classification based on spectral correlation analysis and SVM in cognitive radio," in *Advanced Information Networking and Applications, 2008. AINA 2008. 22nd International Conference on*, Mar. 2008, pp. 883 –887.

[104] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *J. American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, Dec. 2006.

[105] M. Bkassiny, S. K. Jayaweera, and Y. Li, "Multidimensional Dirichlet process-based non-parametric signal classification for autonomous self-learning cognitive radios," *IEEE Trans. Wireless Commun.*, May 2012, [In review].

[106] J. Unnikrishnan and V. V. Veeravalli, "Algorithms for dynamic spectrum access with learning for cognitive radio," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 750 –760, Feb. 2010.

[107] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 589–600, Apr. 2007.

[108] Q. Zhao, L. Tong, and A. Swami, "Decentralized cognitive MAC for dynamic spectrum access," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, Nov. 2005, pp. 224–232.

[109] S. K. Jayaweera and C. Mosquera, "A dynamic spectrum leasing (DSL) framework for spectrum sharing in cognitive radio networks," in *43rd Annual Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, Nov. 2009.

[110] K. Hakim, S. Jayaweera, G. El-Howayek, and C. Mosquera, "Efficient dynamic spectrum sharing in cognitive radio networks: Centralized dynamic spectrum leasing (C-DSL)," *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2956 –2967, Sep. 2010.

[111] B. Latifa, Z. Gao, and S. Liu, "No-regret learning for simultaneous power control and channel allocation in cognitive radio networks," in *Computing, Communications and Applications Conference (ComComAp '12)*, Hong Kong, China, Jan. 2012, pp. 267 –271.

[112] Z. Han, C. Pandana, and K. Liu, "Distributive opportunistic spectrum access for cognitive radio using correlated equilibrium and no-regret learning," in *IEEE Wireless Commun. and Networking Conference (WCNC '07)*, Hong Kong, China, Mar. 2007, pp. 11 –15.

[113] Q. Zhu, Z. Han, and T. Basar, "No-regret learning in collaborative spectrum sensing with malicious nodes," in *IEEE International Conference on Communications (ICC '10)*, Cape Town, South Africa, May 2010, pp. 1 –6.

[114] D. Pados, P. Papantoni-Kazakos, D. Kazakos, and A. Koyiantis, "Online threshold learning for Neyman-Pearson distributed detection," *IEEE Trans. Syst. Man Cybern.*, vol. 24, no. 10, pp. 1519 –1531, Oct. 1994.

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

[116] M. D. Escobar, "Estimating normal means with a Dirichlet process prior," *J. American Statistical Association*, vol. 89, no. 425, pp. 268–277, Mar. 1994. [Online]. Available: http://www.jstor.org/stable/2291223

[117] C. Watkins, "Learning from delayed rewards," Ph.D. dissertation, University of Cambridge, United Kingdom, 1989.

[118] H. Li, "Multi-agent Q-learning of channel selection in multi-user cognitive radio systems: A two by two case," in *IEEE International Conference on Systems, Man and Cybernetics (SMC '09)*, San Antonio, TX, Oct. 2009, pp. 1893 –1898.

[119] J. Peters and S. Schaal, "Policy gradient methods for robotics," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (2006)*, Beijing, China, Oct. 2006, pp. 2219 –2225.

[120] M. Riedmiller, J. Peters, and S. Schaal, "Evaluation of policy gradient methods and variants on the cart-pole benchmark," in *IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL '07)*, Honolulu, HI, Apr. 2007, pp. 254 –261.

[121] D. Fudenberg and J. Tirole, *Game Theory*. MIT Press, 1991.

[122] P. Zhou, W. Yuan, W. Liu, and W. Cheng, "Joint power and rate control in cognitive radio networks: A game-theoretical approach," in *Proc. IEEE International Conference on Communications (ICC'08)*, May 2008, pp. 3296 – 3301.

[123] A. R. Fattahi, F. Fu, M. V. D. Schaar, and F. Paganini, "Mechanism-based resource allocation for multimedia transmission over spectrum agile wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 3, no. 25, pp. 601–612, Apr. 2007.

[124] O. Ileri, D. Samardzija, and N. B. Mandayam, "Demand responsive pricing and competitive spectrum allocation via a spectrum server," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, Nov. 2005, pp. 194–202.

[125] Y. Zhao, S. Mao, J. Neel, and J. Reed, "Performance evaluation of cognitive radios: Metrics, utility functions, and methodology," *Proc. IEEE*, vol. 97, no. 4, pp. 642–659, Apr. 2009.

[126] J. Neel, R. M. Buehrer, B. H. reed, and R. P. Gilles, "Game theoretic analysis of a network of cognitive radio," in *45th Midwest Symp. on Circuits and Systems*, vol. 3, Aug. 2002, pp. III–409–III–412.

[127] M. R. Musku and P. Cotae, "Cognitive radio: Time domain spectrum allocation using game theory," in *IEEE Int. Conf. on System and Systems Engineering (SoSE)*, Apr. 2007, pp. 1–6.

[128] W. Wang, Y. Cui, T. Peng, and W. Wang, "Noncooperative power control game with exponential pricing for cognitive radio network," in *IEEE 65th Vehicular Technology Conf. (VTC)-Spring*, Apr. 2007, pp. 3125–3129.

[129] J. Li, D. Chen, W. Li, and J. Ma, "Multiuser power and channel aloocation algorithm in cognitive radio," in *Int. Conf. on Parallel Processing, (ICPP)*, Sep. 2007, pp. 72–72.

[130] Z. Ji and K. J. R. Liu, "Cognitive radios for dynamic spectrum access- dynamic spectrum sharing: A game theoretical overview," *IEEE Commun. Mag.*, vol. 45, no. 5, pp. 88–94, May 2007.

[131] N. Nie and C. Comaniciu, "Adaptive channel allocation spectrum etiquette for cognitive radio networks," in *1st IEEE Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, Nov. 2005, pp. 269–278.

[132] R. G. Wendorf and H. Blum, "A channel-change game for multiple interfering cognitive wireless networks," in *Military Communi. Conf. (MILCOM)*, Oct. 2006, pp. 1–7.

[133] J. Li, D. Chen, W. Li, and J. Ma, "Multiuser power and channel allocation algorithm in cognitive radio," in *International Conference on Parallel Processing (ICPP '07)*, XiAn, China, Sep. 2007, p. 72.

[134] X. Zhang and J. Zhao, "Power control based on the asynchronous distributed pricing algorithm in cognitive radios," in *IEEE Youth Conference on Information Computing and Telecommunications (YC-ICT '10)*, Beijing, China, Nov. 2010, pp. 69 –72.

[135] L. Pillutla and V. Krishnamurthy, "Game theoretic rate adaptation for spectrum-overlay cognitive radio networks," in *IEEE Global Telecommunications Conference (GLOBECOM '08)*, New Orleans, LA, Dec. 2008, pp. 1 –5.

[136] H. Li, Y. Liu, and D. Zhang, "Dynamic spectrum access for cognitive radio systems with repeated games," in *IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS '10)*, Beijing, China, June 2010, pp. 59 –62.

[137] S. K. Jayaweera and M. Bkassiny, "Learning to thrive in a leasing market: an auctioning framework for distributed dynamic spectrum leasing (D-DSL)," in *IEEE Wireless Commun. & Networking Conference (WCNC '11)*, Cancun, Mexico, Mar. 2011.

[138] L. Chen, S. Iellamo, M. Coupechoux, and P. Godlewski, "An auction framework for spectrum allocation with interference constraint in cognitive radio networks," in *IEEE INFOCOM '10*, San Diego, CA, Mar. 2010, pp. 1 –9.

[139] G. Iosifidis and I. Koutsopoulos, "Challenges in auction theory driven spectrum management," *IEEE Commun. Mag.*, vol. 49, no. 8, pp. 128 –135, Aug. 2011.

[140] F. Fu and M. van der Schaar, "Stochastic game formulation for cognitive radio networks," in *3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '08)*, Chicago, IL, Oct. 2008, pp. 1 –5.

[141] 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 Trans. Wireless Commun.*, vol. 11, no. 4, pp. 1380 –1391, Apr. 2012.

[142] T. Ferguson, "A Bayesian analysis of some nonparametric problems," *The Annals of Statistics*, vol. 1, pp. 209–230, 1973.

[143] D. Blackwell and J. MacQueen, "Ferguson distribution via Polya urn schemes," *The Annals of Statistics*, vol. 1, pp. 353–355, 1973.

[144] M. Jordan. (2005) Dirichlet processes, Chinese restaurant processes and all that. [Online]. Available: http://www.cs.berkeley.edu/ jordan/nips-tutorial05.ps

[145] N. Tawara, S. Watanabe, T. Ogawa, and T. Kobayashi, "Speaker clustering based on utterance-oriented Dirichlet process mixture model," in *12th Annual Conference of the International Speech Communication Association (ISCA '11)*, Florence, Italy, Aug. 2011, pp. 2905–2908.

[146] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953. [Online]. Available: http://dx.doi.org/10.1063/1.1699114

[147] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721 –741, nov. 1984.

[148] N. Shetty, S. Pollin, and P. Pawelczak, "Identifying spectrum usage by unknown systems using experiments in machine learning," in *IEEE Wireless Communications and Networking Conference (WCNC '09)*, Budapest, Hungary, Apr. 2009, pp. 1 –6.

[149] G. Yu, R. Huang, and Z. Wang, "Document clustering via Dirichlet process mixture model with feature selection," in *Proc. 16th ACM SIGKDD International conference on Knowledge Discovery and Data mining (KDD '10)*. New York, NY, USA: ACM, 2010, pp. 763–772. [Online]. Available: http://doi.acm.org/10.1145/1835804.1835901

[150] S. S. Haykin, *Neural networks : A Comprehensive Foundation*, 2nd ed. Prentice Hall, Jul. 1999.

[151] N. Baldo and M. Zorzi, "Learning and adaptation in cognitive radios using neural networks," in *5th IEEE Consumer Communications and Networking Conference (CCNC '08)*, Jan. 2008, pp. 998 –1003.

[152] N. Baldo, B. Tamma, B. Manojt, R. Rao, and M. Zorzi, "A neural network based cognitive controller for dynamic channel selection," in *IEEE International Conference on Communications (ICC '09)*, June 2009, pp. 1 –5.

[153] Y.-J. Tang, Q.-Y. Zhang, and W. Lin, "Artificial neural network based spectrum sensing method for cognitive radio," in *6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM '10)*, Sep. 2010, pp. 1 –4.

[154] M. I. Taj and M. Akil, "Cognitive radio spectrum evolution prediction using artificial neural networks based multivariate time series modeling," *Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless), 11th European*, pp. 1 –6, Apr. 2011.

[155] J. Popoola and R. van Olst, "A novel modulation-sensing method," *IEEE Veh. Technol. Mag.*, vol. 6, no. 3, pp. 60 –69, Sep. 2011.

[156] M. Han, J. Xi, S. Xu, and F.-L. Yin, "Prediction of chaotic time series based on the recurrent predictor neural network," *IEEE Trans. Signal Process.*, vol. 52, no. 12, pp. 3409 – 3416, dec. 2004.

[157] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[158] T. Atwood, "RF channel characterization for cognitive radio using support vector machines," Ph.D. dissertation, University of New Mexico, Nov. 2009.

[159] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. fifth annual workshop on Computational Learning Theory*, ser. COLT '92. New York, NY, USA: ACM, 1992, pp. 144–152. [Online]. Available: http://doi.acm.org/10.1145/130385.130401

[160] M. Martinez-Ramon and C. G. Christodoulou, *Support Vector Machines for Antenna Array Processing and Electromagnetics*, 1st ed., C. A. Balanis, Ed. USA: Morgan and Claypool Publishers, 2006.

[161] H. Hu, J. Song, and Y. Wang, "Signal classification based on spectral correlation analysis and SVM in cognitive radio," in *22nd International Conference on Advanced Information Networking and Applications (AINA '08)*, Okinawa, Japan, Mar. 2008, pp. 883 –887.

[162] G. Xu and Y. Lu, "Channel and modulation selection based on support vector machines for cognitive radio," in *International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '06)*, Sep. 2006, pp. 1 –4.

[163] L. Hai-Yuan and J.-C. Sun, "A modulation type recognition method using wavelet support vector machines," in *2nd International Congress on Image and Signal Processing (CISP '09)*, Oct. 2009, pp. 1 –4.

[164] Z. Yang, Y.-D. Yao, S. Chen, H. He, and D. Zheng, "MAC protocol classification in a cognitive radio network," in *19th Annual Wireless and Optical Communications Conference (WOCC '10)*, May 2010, pp. 1 –5.

[165] M. Petrova, P. Ma andho andnen, and A. Osuna, "Multi-class classification of analog and digital signals in cognitive radios using support vector machines," in *7th International Symposium on Wireless Communication Systems (ISWCS '10)*, Sep. 2010, pp. 986 –990.

[166] D. Zhang and X. Zhai, "SVM-based spectrum sensing in cognitive radio," in *7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '11)*, Sep. 2011, pp. 1 –4.

[167] T. D. Atwood, M. Martnez-Ramon, and C. G. Christodoulou, "Robust support vector machine spectrum estimation in cognitive radio," in *Proc. 2009 IEEE International Symposium on Antennas and Propagation and USNC/URSI National Radio Science Meeting*, 2009.

[168] Z. Sun, G. Bradford, and J. Laneman, "Sequence detection algorithms for PHY-layer sensing in dynamic spectrum access networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 97 –109, Feb. 2011.

[169] D. Cabric, "Addressing feasibility of cognitive radios," *IEEE Signal Processing Mag.*, vol. 25, no. 6, pp. 85 –93, Nov. 2008.

[170] Z. Han, R. Fan, and H. Jiang, "Replacement of spectrum sensing in cognitive radio," *IEEE Trans. Wireless Commun.*, vol. 8, no. 6, pp. 2819 –2826, June 2009.

[171] S. Jha, U. Phuyal, M. Rashid, and V. Bhargava, "Design of OMC-MAC: An opportunistic multi-channel MAC with QoS provisioning for distributed cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 10, pp. 3414 –3425, Oct. 2011.

[172] B. Wang, K. Liu, and T. Clancy, "Evolutionary game framework for behavior dynamics in cooperative spectrum sensing," in *IEEE Global Telecommunications Conference (IEEE GLOBECOM '08)*, Dec. 2008, pp. 1 –5.

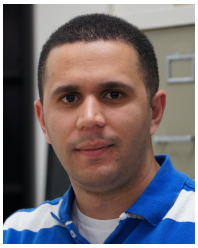
[173] E. C. Y. Peh, Y.-C. Liang, Y. L. Guan, and Y. Zeng, "Power control in cognitive radios under cooperative and non-cooperative spectrum sensing," *IEEE Trans. Wireless Commun.*, vol. 10, no. 12, pp. 4238 –4248, Dec. 2011.

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

[175] L. Chen, S. Iellamo, M. Coupechoux, and P. Godlewski, "An auction framework for spectrum allocation with interference constraint in cognitive radio networks," in *Proc. IEEE INFOCOM '10*, Mar. 2010, pp. 1 –9.

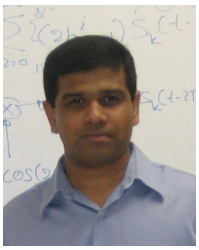
[176] M. Haddad, S. Elayoubi, E. Altman, and Z. Altman, "A hybrid approach for radio resource management in heterogeneous cognitive networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 4, pp. 831 –842, Apr. 2011.

[177] S. Buljore, M. Muck, P. Martigne, P. Houze, H. Harada, K. Ishizu, O. Holland, A. Mikhailovic, K. A. Tsagkariss, O. Sallent, M. S. G. Clemo, V. Ivanov, K. Nolte, and M. Stametalos, "Introduction to IEEE p1900.4 activities," *IEICE Trans. Commun.*, vol. E91-B, no. 1, 2008.



Mario Bkassiny (S'06) received the B.E. degree in Electrical Engineering with High Distinction and the M.S. degree in Computer Engineering from the Lebanese American University, Lebanon, in 2008 and 2009, respectively. He is currently working towards his PhD degree in Electrical Engineering at the Communication and Information Sciences Laboratory (CISL), Department of Electrical and Computer Engineering at the University of New Mexico, Albuquerque, NM, USA. His current research interests are in cognitive radios, distributed learning and reasoning, cognitive and cooperative communications, machine learning and dynamic spectrum leasing (DSL).

Yang Li received the B.E. degree in Electrical Engineering from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2005 and the M.S. degree in Electrical Engineering from New Mexico Institute of Mining and Technology, Socorro, New Mexico, USA in 2009. He is currently working towards his PhD degree in Electrical Engineering at the Communication and Information Sciences Laboratory (CISL), Department of Electrical and Computer Engineering at the University of New Mexico, Albuquerque, NM, USA. His current research interests are in cognitive radios, spectrum sensing, cooperative communications, and dynamic spectrum access (DSA).



Sudharman K. Jayaweera (S'00, M'04, SM'09) was born in Matara, Sri Lanka. He received the B.E. degree in Electrical and Electronic Engineering with First Class Honors from the University of Melbourne, Australia, in 1997 and M.A. and PhD degrees in Electrical Engineering from Princeton University in 2001 and 2003, respectively. He is currently an associate Professor in Electrical Engineering at the Department of Electrical and Computer Engineering at University of New Mexico, Albuquerque, NM. Dr. Jayaweera held an Air Force Summer Faculty Fellowship at the Air Force Research Laboratory, Space Vehicles Directorate (AFRL/RVSV) during the summers of 2009-2011.

Dr. Jayaweera is currently an associate editor of IEEE Trans. Vehicular Technology and EURASIP Journal on Advances in Signal Processing. He has also served as a member of the Technical Program Committees of numerous IEEE conferences including ICC (2010-2012), Globecom (2006, 2008, 2009,2011), WCNC (2011, 2012) and PIMRC (2011, 2012). His current research interests include cooperative and cognitive communications, information theory of networked-control systems, control and optimization in smart-grid, machine learning techniques for cognitive radios, statistical signal processing and wireless sensor networks.