EURASIP Journal on Wireless Communications and Networking
a SpringerOpen Journal

RESEARCH Open Access

Sensing time and power allocation for cognitive radios using distributed Q-learning

Olivier van den Biggelaar*, Jean-Michel Dricot, Philippe De Doncker and François Horlin

Abstract

In cognitive radios systems, the sparse assigned frequency bands are opened to secondary users, provided that the aggregated interferences induced by the secondary transmitters on the primary receivers are negligible. Cognitive radios are established in two steps: the radios firstly sense the available frequency bands and secondly communicate using these bands. In this article, we propose two decentralized resource allocation Q-learning algorithms: the first one is used to share the sensing time among the cognitive radios in a way that maximize the throughputs of the radios. The second one is used to allocate the cognitive radio powers in a way that maximizes the signal on interference-plus-noise ratio (SINR) at the secondary receivers while meeting the primary protection constraint. Numerical results show the convergence of the proposed algorithms and allow the discussion of the exploration strategy, the choice of the cost function and the frequency of execution of each algorithm.

1. Introduction

The scarcity of available radio spectrum frequencies, densely allocated by the regulators, represents a major bottleneck in the deployment of new wireless services. Cognitive radios have been proposed as a new technology to overcome this issue [1]. For cognitive radio use, the assigned frequency bands are opened to secondary users, provided that interference induced on the primary licensees is negligible. Cognitive radios are established in two steps: the radios firstly sense the available frequency bands and secondly communicate using these bands.

To tackle the fading phenomenon–an attenuation of the received power due to destructive interferences between the multiple interactions of the emitted wave with the environment–when sensing the frequency spectrum, cooperative spectrum sensing has been proposed to take advantage of the spatial diversity in wireless channels [2,3]. In cooperative spectrum sensing, the secondary cognitive nodes send the results of their individual observations of the primary signal to a base station through specific control channels. The base station then combines the received information in order to make a decision about the primary network presence. Each cognitive node observes the primary signal during a certain sensing time, which should be chosen high enough to ensure the correct detection of the primary emitter but low enough so that the node has still enough time to communicate. In literature [4,5], the sensing times used by the cognitive nodes are generally assumed to be identical and allocated by a central authority. In [6], the sensing performance of a network of independent cognitive nodes that individually select their sensing times is analyzed using evolutionary game theory.

It is generally considered in literature that the secondary users can only transmit if the primary network is inactive or if the secondary users are located outside a keep-out region surrounding the primary transmitter, or equivalently, if the secondary users generate an interference inferior to a given threshold on a so called *protection contour* surrounding the primary transmitter [7,8]. However, multiple simultaneously transmitting secondary users may individually meet the protection contour constraint while collectively generating an aggregated interference that exceeds the acceptable threshold. In [7], the effect of aggregated interference caused by IEEE 802.22 secondary users on primary DTV receivers is analyzed. In [9], the aggregated interference generated by a large-scale secondary network is modeled and the impact of the secondary network density on the sensing requirements is investigated. In [10], a decentralized power allocation Q-learning algorithm is proposed to protect the primary network from harmful aggregated interference. The proposed algorithm removes the need

* Correspondence: ovdbigge@ulb.ac.be
Université Libre de Bruxelles (ULB), Avenue F. D. Roosevelt 50, B-1050 Brussels, Belgium

for a central authority to allocate the powers in the secondary network and therefore minimizes the communication overhead. The cost functions used by the algorithm are chosen so that the aggregated interference constraint is exactly met on the protection contour. Unfortunately, the cost functions do not take into account the preferences of the secondary network.

This article aims to illustrate the potential of Q-learning for cognitive radio systems. For this purpose two decentralized Q-learning algorithm are presented to solve the allocation problems that appear during the sensing phase on the one hand and during the communication phase on the other hand. The first algorithm allows to share the sensing times among the cognitive radios in a way that maximize the throughputs of the radios. The second algorithm allows to allocate the secondary user powers in a way that maximize the signal on interference-plus-noise ratio (SINR) at the secondary receivers while meeting the primary protection constraint. The agents self-adapt by directly interacting with the environment in real time and by properly utilizing their past experience. They aim to distributively learn an optimal strategy to maximize their throughputs or their SINRs.

Reinforcement learning algorithms such as Q-learning are particularly efficient in applications where reinforcement information (i.e., cost or reward) is provided after an action is performed in the environment [11]. The sensing time and power allocation problems both allow for the easy definition of such information. In this article, we make the assumption that no information is exchanged between the agents for each of the two problems. As a result, many traditional multi-agent reinforcement learning algorithms like fictitious play and Nash-Q learning cannot be used [12], which justifies the use of multi-agent Q-learning in this article to solve the sensing time and power allocation problems.

This distributed allocation of the sensing times and the node powers presents several advantages compared to a centralized allocation [10]: (1) robustness of the system towards a variation of parameters (such as the gains of the sensing channels), (2) maintainability of the system thanks to the modularity of the multiple agents and (3) scalability of the system as the need for control communication is minimized: on the one hand there is no need for a central authority to send the result of a centralized allocation to the multiple nodes and on the other hand these nodes do not have to send their specific parameters (sensing SNRs and data rates for the sensing time allocation, space coordinates for the power allocation problem). In addition, a centralized allocation is not a trivial operation as the sensing time and the power allocation problems are both essentially multi-criteria problems where multiple objective function to maximize can be defined (e.g., the sum of the individual rewards to aim for a global optimum or the minimum individual reward to guarantee more fairness).

The rest of this article is organized as follows: in Section 2, we formulate the problems of sensing time allocation in the secondary network. In Section 3, we formulate the problem of power allocation in the secondary network. In Section 4, we present the decentralized Q- learning algorithms used to solve the sensing time allocation problem and the power allocation problem. In Section 5, we present numerical results allowing the discussion of the performance of the Q-learning algorithms for different exploration strategies, cost functions and execution frequencies.

2. Sensing time allocation problem formulation

2.1. Cooperative spectrum sensing

The licensed band is assumed to be divided into N sub-bands, and each secondary user is assumed to communicate in one of the N sub-bands when the primary user is absent. When it is present, the primary network is assumed to use all N sub-bands for its communications. Therefore, the secondary user can jointly sense the primary network presence on these sub-bands and report their observations via a narrowband control channel.

We consider a cognitive radio cell made of $N + 1$ nodes including a central base station. Each node j performs an energy detection of the received signal using M_j samples [13,14]. The observed energy value at the j^{th} node is given by the random variable:

$$Y_j = \begin{cases} \sum_{i=1}^{M_j} n_{ji}^2, & \text{under } H_0 \\ \sum_{i=1}^{M_j} (s_{ji} + n_{ji})^2, & \text{under } H_1 \end{cases}$$

where s_{ji} and n_{ji} denote the received primary signal and additive white noise at the ith sample of the jth cognitive radio, respectively, ($1 \le j \le N$, $1 \le i \le M_j$). These samples are assumed to be real without loss of generality. H_0 and H_1 represent the hypotheses associated to primary signal absence and presence, respectively. In the distributed detection problem, the coordinator node receives information from each of the N nodes (e.g., the communicated Y_j) and must decide between the two hypotheses.

We assume that the instantaneous noise at each node n_{ji} can be modeled as a zero-mean Gaussian random variable with unit variance $n_{ji} \sim \mathcal{N}(0, 1)$. Let γ_j be the signal-to-noise ratio (SNR) computed at the jth node, defined as $\gamma_j \sim \frac{1}{M_j} \sum_{i=1}^{M_j} s_{ji}^2$.

Since $n_{ji} \sim \mathcal{N}(0, 1)$, the random variable Y_j can be expressed as:

$$Y_j \sim \begin{cases} \chi^2_{M_j}, & \text{under } H_0 \\ \chi^2_{M_j}(\lambda_j), & \text{under } H_1 \end{cases}$$

where $\chi^2_{M_j}$ denotes a central chi-squared distribution with M_j degrees of freedom and $\lambda_j = M_j\gamma_j$ is the non-centrality parameter. Furthermore, if M_j is large, the Central Limit theorem gives [15]:

$$Y_j \sim \begin{cases} \mathcal{N}(M_j, 2M_j), & \text{under } H_0 \\ \mathcal{N}(M_j(1+\gamma_j), 2M_j(1+2\gamma_j)), & \text{under } H_1 \end{cases} \quad (1)$$

From (1), it can be shown that the false alarm probability $P_{F_j} = \Pr\{Y_j > \lambda | H_0\}$ is given by:

$$P_{F_j} = Q\left(\frac{\lambda - M_j}{\sqrt{2M_j}}\right) \quad (2)$$

and the detection probability $P_{D_j} = \Pr\{Y_j > \lambda | H_1\}$ is given by:

$$P_{D_j} = Q\left(\frac{\lambda - M_j(1+\gamma_j)}{\sqrt{2M_j(1+2\gamma_j)}}\right), \quad (3)$$

where $Q(x) = \int_x^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$.

By combining Equations (2) and (3), the false alarm probability can be expressed with respect to the detection probability:

$$P_{F_j} = Q\left(Q^{-1}(P_{D_j})\sqrt{(1+2\gamma_j)} + \gamma_j\sqrt{\frac{M_j}{2}}\right). \quad (4)$$

where $Q^{-1}(x)$ is the inverse function of $Q(x)$.

As illustrated on Figure 1, we consider that every T_H seconds, each node sends a one bit value representing the local hard decision about the primary network presence to the base station. The base station combines the received bits in order to make a global decision for the nodes. The base station decision is sent back to the node as a one bit value. The duration of the communication with the base station is assumed to be negligible compared to the duration T_H of a time slot. In this article, we focus on the logical-OR fusion rule at the base station but the other fusion rules could be similarly analyzed. Under the logical-OR fusion rule, the global detection probability P_D (defined as the probability that the coordinator node identifies a time slot as busy when the primary network is present during this time slot) and the global false alarm probability P_F (defined as the probability that the coordinator node identifies a time slot as busy when the primary network is absent during this time slot) depend, respectively, on the local detection probabilities P_{D_j} and false alarm probabilities P_{F_j} [16]:

$$P_D = 1 - \prod_{j=1}^{N}(1 - P_{D_j}), \quad (5)$$

and

$$P_F = 1 - \prod_{j=1}^{N}(1 - P_{F_j}). \quad (6)$$

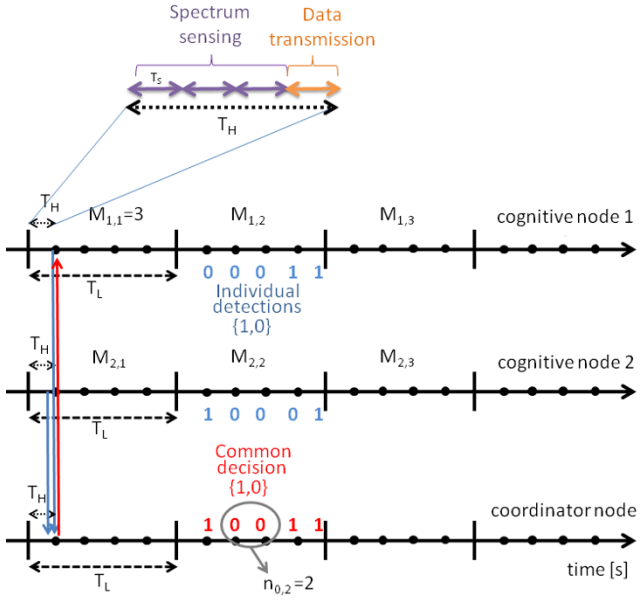

Figure 1 Time diagram of the sensing time allocation algorithm.

Given a target global detection probability $\bar{P}_D$, we thus have:

$$P_{D_j} = 1 - (1 - \bar{P}_D)^{1/N}, \tag{7}$$

and Equation (4) can be rewritten as:

$$P_{F_j} = Q\left(Q^{-1}\left(1 - (1 - \bar{P}_D)^{1/N}\right)\sqrt{(1 + 2\gamma_j)} + \gamma_j\sqrt{\frac{M_j}{2}}\right). \tag{8}$$

2.2. Throughput of a secondary user

The random variable representing the presence of the primary network in each time slot n is denoted $H(n)$ ($H(n) \in \{H_0, H_1\}$) and is assumed to be a Markov Chain characterized by a transition matrix $[p_{uv}]$. It is assumed that the probability p_{01} of the primary network apparition is small compared to the probability p_{00}. As a result, the secondary users can decide to communicate or not during a time slot based on the result of their sensing in the previous time slot while limiting the probability of interference with the primary network.

A secondary user performs data transmission during the time slots that have been identified as free by the base station. In each of these time slots, M_j T_S seconds are used by the secondary user to sense the spectrum, where T_S denotes the sampling period. The remaining T_H - M_jT_S seconds are used for data transmission. The secondary user average throughput R_j is given by the sum of the throughput obtained when the primary network is absent and no false alarm has been generated by the base station plus the throughput obtained when the primary network is present but has not been detected by the base station [6]:

$$\begin{aligned} R_j = &\frac{T_H - M_jT_S}{T_H}\pi_{H_0}(1 - P_F)p_{00}C_{H_0,j} \\ &+ \frac{T_H - M_jT_S}{T_H}(1 - \pi_{H_0})(1 - \bar{P}_D)p_{10}C_{H_0,j} \\ &+ \frac{T_H - M_jT_S}{T_H}(1 - \pi_{H_0})(1 - \bar{P}_D)p_{11}C_{H_1,j} \\ &+ \frac{T_H - M_jT_S}{T_H}\pi_{H_0}(1 - P_F)p_{01}C_{H_1,j} \end{aligned} \tag{9}$$

where $\pi_{H_0} = \lim_{n\to\infty}\frac{\sum_{v=1}^{n}\mathbf{1}_{H(v)=H_0}}{n}$ denote the stationary probability of the primary network absence, $C_{H_0,j}$ represents the data rate of the secondary user under H_0 and $C_{H_1,j}$ represents the data rate of the secondary user under H_1. The target detection probability $\bar{P}_D$ is required to be close to 1 since the cognitive radios should not interfere with the primary network; moreover, π_{H_0} is usually close to 1, $C_{H_1,j} \ll C_{H_0,j}$ due to the interference from the primary network [6] and it is assumed that $p_{00} \geq p_{10}$. Therefore, (9) can be approximated by:

$$R_j \approx \frac{T_H - M_jT_S}{T_H}\pi_{H_0}(1 - P_F)p_{00}C_{H_0,j} \tag{10}$$

2.3. Sensing time allocation problem

Equations (6), (8), and (10) show that there is a tradeoff for the choice of the sensing window length M_j: on the one hand, if M_j is high then the user j will not have enough time to perform his data transmission and R_j will be low. On the other hand, if all the users use low M_j values, then the global false alarm probability in (10) will be high and all the average throughputs will be low.

The sensing time allocation problem consists in finding the optimal sensing window length $\{M_1, \ldots, M_N\}$ that minimize a cost function $f(R_1, \ldots, R_N)$ depending on the secondary throughputs.

In this article, the following cost function is considered:

$$f(R_1, \ldots, R_N) = \sum_{j=1}^{N}(R_j - \bar{R}_j)^2 \tag{11}$$

where $\bar{R}_j$ denotes the throughput required by node j.

It is observed that the cost decreases with respect to R_j until R_j reaches the threshold value $\bar{R}_j$, then the cost increases with respect to R_j. This should prevent secondary users from selfishly transmitting with a throughput higher than required, which would reduce the achievable throughputs for the other secondary users.

Although a base station could determine the sensing window lengths that minimize function (11) and send these optimal values to each secondary user, in this article we rely on the secondary users themselves to determine their individual best sensing window length. This decentralized allocation avoids the introduction of signaling overhead in the system.

3. Power allocation problem formulation

We consider a large circular primary cell made up of one central primary emitter and several primary receivers whose positions are unknown. The primary emitter could be a DTV broadcasting station that communicates with multiple passive receivers.

The secondary network uses the same frequency band as the primary network and consists in L adjacent secondary cells. Each secondary cell is made up of one central secondary base station and multiple secondary users. For the sake of simplicity, all the secondary users (SU) are assumed to be located on the line that joins the L base stations (BS) as illustrated on Figure 2. The reader

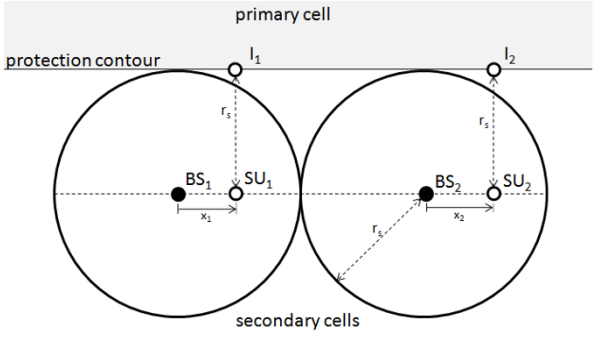

Figure 2 Primary and secondary networks when $L = 2$ adjacent secondary cells are deployed.

is referred to [10] for more realistic assumptions regarding the geometry of the power allocation problem.

In order to protect the primary receivers from receiving harmful interference from the secondary users, a *protection contour* is defined around the primary emitter as a circle on which the received primary SINR must be superior to a given threshold $\mathrm{SINR}^{p}_{\mathrm{Th}}$. The secondary cells are located around the protection contour. As the primary cell ray is assumed to be much larger than the secondary cells ray, the protection contour can be approximated by a line parallel to the secondary base stations line.

The secondary network is assumed to follow a Time Division Multiple Access (TDMA) scheme, so that at each time only one secondary user SU_l communicates with its base station BS_l in cell l ($l \in \{1, \ldots, L\}$). The difference between SU_l and BS_l abscissa is denoted x_l. The point on the protection contour whose distance with SU_l is minimal is denoted I_l. We assume that each cell l deploys sensors on the protection contour so that it is able to measure the primary network SINR at the point I_l , denoted SINR^{p}_{l}.

In this article, the analysis is focused on the interference generated by the upstream transmissions of the secondary users. It is assumed that the secondary SINR at each base station l, denoted SINR^{s}_{l}, needs to be superior to a given threshold $\mathrm{SINR}^{s}_{\mathrm{Th}}$ for the secondary communication to be reliable.

The power allocation problem consists in finding the optimal secondary users transmission powers $\{P_1, \ldots, P_L\}$ that minimize a cost function $f\left(\mathrm{SINR}^{s}_{1}, \ldots, \mathrm{SINR}^{s}_{L}\right)$ depending on the secondary SINRs, under the constraints that

$$\mathrm{SINR}^{p}_{l} \geq \mathrm{SINR}^{p}_{\mathrm{Th}} \quad \forall l \in \{1, \ldots, L\} \tag{12}$$

In this article, the following cost function is considered:

$$f\left(\mathrm{SINR}^{s}_{1}, \ldots, \mathrm{SINR}^{s}_{L}\right) = \sum_{l=1}^{L} \left(\mathrm{SINR}^{s}_{l} - \mathrm{SINR}^{s}_{\mathrm{Th}}\right)^2 \tag{13}$$

It is observed that the cost decreases with respect to SINR^{s}_{l} until SINR^{s}_{l} reaches the threshold value $\mathrm{SINR}^{s}_{\mathrm{Th}}$, then the cost increases with respect to SINR^{s}_{l}. This should prevent secondary users from selfishly transmitting with a power higher than required, which would remove transmission opportunities for other secondary users.

The primary SINRs in Equation (12) are given by:

$$\mathrm{SINR}^{p}_{l} = \frac{P^{p}}{\sigma^2 + \sum_{k=1}^{L} P_k h^{\mathrm{SU}_k}_{I_l}}$$

where P^{p} is the power that is received on the protection contour from the primary transmitter, σ^2 is the noise power and $h^{\mathrm{SU}_k}_{\mathrm{I}_l}$ is the link gain between SU_k and the point I_l on the protection contour.

The secondary SINRs in Equation (13) are given by:

$$\mathrm{SINR}^{s}_{l} = \frac{P_l h^{\mathrm{SU}_l}_{\mathrm{BS}_l}}{\sigma^2 + \sum_{k=1, k\neq l}^{N} P_k h^{\mathrm{SU}_k}_{\mathrm{BS}_l}}$$

where $h^{\mathrm{SU}_l}_{\mathrm{BS}_l}$is the link gain between SU_l and BS_l.

In this article, we consider free space path loss. Therefore, the link gains are computed as follows:

$$h^{\mathrm{SU}_k}_{\mathrm{I}_l} = \left(\frac{4\pi f_c}{c}\sqrt{r_s^2 + (2(k-l)r_s - x_l + x_k)^2}\right)^{-2} \tag{14}$$

$$h^{\mathrm{SU}_k}_{\mathrm{BS}_l} = \left(\frac{4\pi f c}{c}(2(k-l)r_s + x_k\right)^{-2} \tag{15}$$

where r_s is the ray of the secondary cells, f_c is the transmission frequency and c is the speed of light in vacuum.

4. Learning algorithm

4.1. Q-learning algorithm

In this article, we use two multi-agent Q-learning algorithms. The first one is used to allocate the secondary user sensing times and the second one is used to allocate the secondary user transmission powers. In the sensing time allocation algorithm, each secondary user is an agent that aims to learn an optimal sensing time allocation policy for itself. In the power allocation algorithm, each secondary base station is an agent that aims to learn an optimal power allocation policy for its cell.

Q-learning implementation requires the environment to be modeled as a finite-state discrete-time stochastic

system. The set of all possible states of the environment is denoted $\mathcal{S}$. At each learning iteration, the agent that executes the learning algorithm performs an action chosen from the finite set $\mathcal{A}$ of all possible actions. Each learning iteration consists in the following sequence:

1) The agent senses the state $s \in \mathcal{S}$ of the environment
2) Based on s and its accumulated knowledge, the agent chooses and performs an action $a \in \mathcal{A}$.
3) Because of the performed action, the state of the environment is modified. The new state is denoted s' The transition from s to s' generates a cost $c \in \mathbb{R}$ for the agent.
4) The agent uses c and s' to update the accumulated knowledge that made him choose the action a when the environment was in state s.

The Q-learning algorithm keeps a quality information (the Q-value) for every state-action couple (s, a) it has tried. The Q-value $Q_i(s, a)$ represents how high the expected quality of an action a is when the environment is in state s [17]. The following policy is used for the selection of the action a by the agent when the environment is in state s:

$$a = \begin{cases} \arg\max_{\tilde{a} \in \mathcal{A}} Q(s, \tilde{a}), & \text{with probability} \quad 1 - \epsilon \\ \text{random action} \in \mathcal{A}, & \text{with probability} \quad \epsilon \end{cases} \quad (16)$$

where ϵ is the randomness for exploration of the learning algorithm.

The cost c and the new state s' generated by the choice of action a in state s are used to update the Q-value $Q(s, a)$ based on how good the action a was and how good the new optimal action will be in state s'. The update is handled by the following rule:

$$Q(s, a) \leftarrow (1 - \alpha) Q(s, a) + \alpha(-c + \gamma \max_{a' \in \mathcal{A}} Q(s', a')) \quad (17)$$

where α is the learning rate and γ is the discount rate of the algorithm.

The learning rate $\alpha \in [0, 1]$ is used to control the linear blend between the previously accumulated knowledge about the (s, a) couple, $Q(s, a)$, and the newly received quality information $\left(-c + \gamma \max_{a' \in \mathcal{A}} Q(s', a')\right)$. A high value of α gives little importance to previous experience, while a low value of α gives an algorithm that learns slowly as the stored Q-values are easily altered by new information.

The discount rate $\gamma \in [0, 1]$ is used to control how much the success of a later action a' should be brought back to the earlier action a that led to the choice of a'. A high value of γ gives a low importance to the cost of the current action compared to the Q-value of the new state this actions leads to, while a low value of γ would rate the current action almost only based on the immediate reward it provides.

The randomness for exploration $\epsilon \in [0, 1]$ is used to control how often the algorithm should take a random action instead of the best action it knows. A high value of ϵ favors exploration of new good actions over exploitation of existing knowledge, while a low value of ϵ reinforces what the algorithm already knows instead of trying to find new better actions. The exploration-exploitation trade-off is typical of learning algorithms. In this article, we consider online learning (i.e., at every time step the agents should display intelligent behaviors) which requires a low ϵ value.

4.2. Q-Learning implementation for sensing time allocation

Each secondary user is an agent in charge of sensing the environment state, selecting an action according to policy (16), performing this action, sensing the resulting new environment state, computing the induced cost and updating the state-action Q-value according to rule (17). In this section, we specify the states, actions and cost function used to solve the sensing time allocation problem.

At each iteration $t \in \{1, \ldots, K\}$ of the learning algorithm, a secondary user $j \in \{1, \ldots, N\}$ represents the local state $s_{j,t}$ of the environment as follows:

$$s_{j,t} = n_{H_0, t-1} \quad (18)$$

where $n_{H_0, t-1}$ denotes the number of time slots that have been identified as free by the base station during the (t -1)th learning period.

The number of free time slots takes one out of r values:

$$n_{H_0, t} \in \{0, 1, \ldots, r\}.$$

At each iteration t, the action selected by the secondary user j is the duration $M_{j,t}$ of the sensing window to be used during the T_L seconds of the learning iteration t. It is assumed that one learning iteration spans several time slots:

$$T_L = rT_H, r \in \mathbb{N}_0.$$

The optimal value of r will be determined in Section 5. Let s denotes the ratio between the duration of a time slot and the sampling period:

$$T_H = sT_S, s \in \mathbb{N}_0,$$

during each learning period t, the sensing window length takes one out of $s + 1$ values:

$$M_{j,t} \in \{0, 1, \ldots, s\}. \tag{19}$$

In this article, we compare the performances of the sensing time allocation system for two different cost functions $c_{j,\,t}$. We firstly define a *competitive cost function* in which the cost decreases if the average throughput realized by node j increases:

$$c_{j,t} = -\hat{R}_{j,t} \tag{20}$$

where $\hat{R}_{j,t}$ denotes the average throughput $\hat{R}_{j,t}$ realized by node j during the learning period t:

$$\hat{R}_{j,t} = \frac{n_{H_0,t}(T_H - M_{j,t}T_S)}{T_L} C_{H_0,j} \tag{21}$$

$$= \frac{n_{H_0,t}}{r}\left(1 - \frac{M_{j,t}}{s}\right) C_{H_0,j} \tag{22}$$

With this cost function, every node tries to achieve the maximum $\hat{R}_{j,t}$ with no consideration for the other nodes in the secondary network. We secondly define a *cooperative cost function* in which the cost decreases if the difference between the realized average throughput and the required average throughput decreases:

$$c_{j,t} = \left(\hat{R}_{j,t} - \bar{R}_j\right)^2 \tag{23}$$

This last cost function penalizes the actions that lead to a realized average throughput that is higher than required, which should help the disadvantaged nodes (i.e., the nodes that have a low data rate $C_{H_0,j}$) to achieve the required average throughput.

4.3. Q-Learning implementation for distributed power allocation

Each secondary BS is an agent in charge of sensing the environment state, selecting an action according to policy (16), performing this action, sensing the resulting new environment state, computing the induced cost and updating the state-action Q-value according to rule (17). In this section, we specify the states, actions, and cost function used to solve the power allocation problem.

At each iteration $t \in \{1, \ldots, K\}$ of the learning algorithm, a base station $l \in \{1, \ldots, L\}$ represents the local state $s_{l,\,t}$ of the environment as the following triplet:

$$s_{l,t} = \{x_{l,t}, P_{l,t}, I_{l,t}\} \tag{24}$$

where $x_{l,\,t}$ is the local coordinate of the currently transmitting secondary user SU_l, $P_{l,\,t}$ is the power currently allocated to this user and $I_{l,\,t} \in \{0, 1\}$ is a binary indicator that specifies whether the measured aggregated interference at the sensor I_l on the protection contour is above or below the acceptable threshold. It is defined as:

$$I_{l,t} = \begin{cases} 1 \text{ if } \mathrm{SINR}^p_{l,t} < \mathrm{SINR}^p_{\mathrm{Th}} \\ 0 \text{ otherwise} \end{cases} \tag{25}$$

For Q-learning implementation the states have to be quantized. Therefore it is assumed that $x_{l,\,t}$ takes one out of the following ξ values:

$$x_{l,t} \in \left\{ \frac{r_s}{\xi}(-(\xi - 1) + 2k) | k \in \{0, \ldots, \xi - 1\} \right\} \tag{26}$$

Similarly, $P_{l,\,t}$ takes one out of the following φ values:

$$P_{l,t} \in \Psi = \left\{ P_{\min} + k\left(\frac{P_{\max} - P_{\min}}{\phi - 1}\right) | k \in \{0, \ldots, \phi - 1\} \right\} \tag{27}$$

where $P_{\min}$ and $P_{\max}$ are the minimum and maximum effective radiated powers (ERP) in dBm.

At each iteration t, the action selected by the base station BS_l is the power to allocate to the currently transmitting secondary user SU_l. The set of all possible actions is therefore given by Equation (27).

In this article, we compare the performances of the power allocation system for two different cost functions $c_{l,\,t}$. We first define a *competitive cost function* in which the cost decreases if the secondary SINR at the base station increases, provided that the aggregated interference generated on the primary protection contour does not exceeds the acceptable level:

$$c_{l,t} = \begin{cases} -\mathrm{SINR}^s_{l,t} \text{ if } \mathrm{SINR}^p_{l,t} \geq \mathrm{SINR}^p_{\mathrm{Th}} \\ +\infty \quad \text{otherwise} \end{cases} \tag{28}$$

where $+\infty$ represents a positive constant that is chosen large enough compared to $\mathrm{SINR}^s_{l,t}$. With this cost function, every agent tries to achieve the maximum SINR^s_l with no consideration for the other secondary cells in the network. Second, we define a *cooperative cost function* in which the cost decreases if the difference between the secondary SINR at the base station and the required secondary SINR threshold decreases, provided that the aggregated interference on the protection contour is acceptable:

$$c_{l,t} = \begin{cases} \left(\mathrm{SINR}^s_{l,t} - \mathrm{SINR}^s_{\mathrm{Th}}\right)^2 \text{ if } \mathrm{SINR}^p_{l,t} \geq \mathrm{SINR}^p_{\mathrm{Th}} \\ +\infty \quad \text{otherwise} \end{cases} \tag{29}$$

where $+\infty$ represents a positive constant that is chosen large enough compared to $\left(\mathrm{SINR}^s_{l,t} - \mathrm{SINR}^s_{\mathrm{Th}}\right)^2$. This last cost function penalizes the actions that lead to a secondary SINR that is higher than required, which should help the disadvantaged secondary cells (i.e., the cells in which the transmission distance $|x_{l,\,t}|$ and/or

the aggregated interference $\sum_{k=1,k\neq l}^{N} P_k h_{\mathrm{BS}_l}^{\mathrm{SU}_k}$ is high) to achieve the required secondary SINR threshold.

In this article, the impact of the frequency of the learning algorithm is also analyzed. If T_{TDMA} denotes the length of a TDMA time slot and T_L denotes the length of a learning iteration, then

$$f = \frac{T_{\mathrm{TDMA}}}{T_L} \tag{30}$$

indicates how many times a learning loop is executed during one TDMA time slot (i.e., for a fixed secondary transmitter SU_l in cell l). It is assumed that every secondary cell uses the same TDMA time slot length T_{TDMA} as well as the same learning iteration length T_L. However, the secondary transmissions as well as the learning iterations are assumed asynchronous, as illustrated on Figure 3.

Finally, three exploration strategies are compared in this article. These three exploration strategies are characterized by the same average randomness for exploration $\bar{\epsilon}$.

The first exploration strategy consists in using a constant ϵ parameter during the K learning iterations:

$$\epsilon_t = \bar{\epsilon} \tag{31}$$

In the second exploration strategy, ϵ decreases linearly between the values $\epsilon_{t=1} = 2\bar{\epsilon}$ and $\epsilon_{t=K} = 0$:

$$\epsilon_t = 2\bar{\epsilon}\left(\frac{K-t}{K-1}\right) \tag{32}$$

In the third exploration strategy, the algorithm does pure exploration during the $\bar{\epsilon}f$ first learning iterations of each TDMA time slot, then pure exploitation during the remaining $(1-\bar{\epsilon})f$ last learning iterations of the time slot (see, Figure 4):

$$\epsilon_t = \begin{cases} 1 \text{ if } t - \left\lfloor \frac{t}{f} \right\rfloor < \bar{\epsilon}f \\ 0 \text{ otherwise} \end{cases} \tag{33}$$

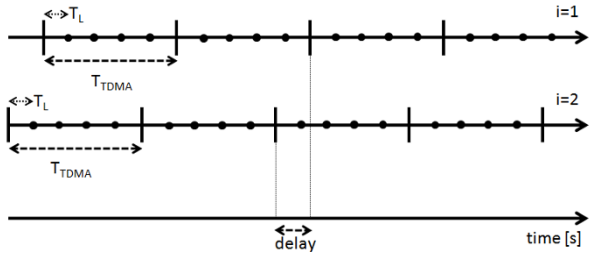

Figure 3 The transmissions and the learning iterations in the multiple secondary cells are assumed asynchronous.

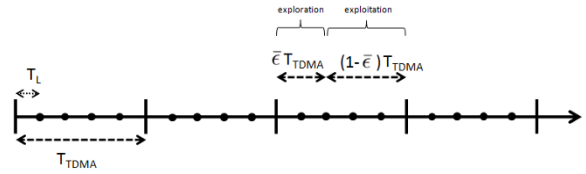

Figure 4 Exploration strategy that consists in doing pure exploration during the $\bar{\epsilon}T_{\mathrm{TDMA}}$ first seconds of each TDMA time slot, then pure exploitation during the remaining $(1-\bar{\epsilon})T_{\mathrm{TDMA}}$ last seconds of the time slot.

Note that for both the sensing time and power allocation problems, the agents have an imperfect knowledge of the state of the environment. The state represented by an agent at each iteration of the Q-learning algorithm is actually an imperfect estimation of the environment state. In this case, the convergence demonstration of single agent Q-learning [18] does not hold. However, multi-agent Q-learning algorithms have been successfully applied in multiple scenarios [11] and in particular to cognitive radios [10,12,19]. Numerical results will show that both Q-learning algorithms presented in this article converge as well.

5. Numerical results

5.1. Sensing time allocation algorithm

Unless otherwise specified, the following simulation parameters are used: we consider $N = 2$ nodes able to transmit at a maximum data rate $C_{H_0,1} = C_{H_0,2} = 0.6$. They each require a data rate $\bar{R}_1 = \bar{R}_2 = 0.1$. One node has a sensing channel characterized by $\gamma_1 = 0$ dB and the second one has a poorer sensing channel characterized by $\gamma_2 = -10$ dB.

It is assumed that the primary network transition probabilities are $p_{00} = 0.9$, $p_{01} = 0.1$, $p_{10} = 0.2$, and $p_{11} = 0.8$. The target detection probability is $\bar{P}_D = 0.95$.

We consider $s = 10$ samples per time slot and $r = 100$ time slots per learning periods. The Q-learning algorithm is implemented with a learning rate $\alpha = 0.5$ and a discount rate $\gamma = 0.7$. The chosen exploration strategy consists in using $\epsilon = 0.1$ during the first $K/2$ iterations and then $\epsilon = 0$ during the remaining $K/2$ iterations.

Figure 5 gives the result of the Q-learning algorithms when no exploration strategy is used ($\epsilon = 0$). It is observed that after 430 iterations, the algorithm converges to $M_1 = M_2 = 4$ which is a sub-optimal solution. The optimal solution obtained by minimizing Equation (11) is $M_1 = 4$, $M_2 = 1$ (as the second node has a low sensing SNR, the first node has to contribute more to the sensing of the primary signal). After convergence, the normalized average throughputs are $\hat{R}_{2,\mathrm{opt}}/C_{H_0,2} = 0.144$ whereas the optimal normalized

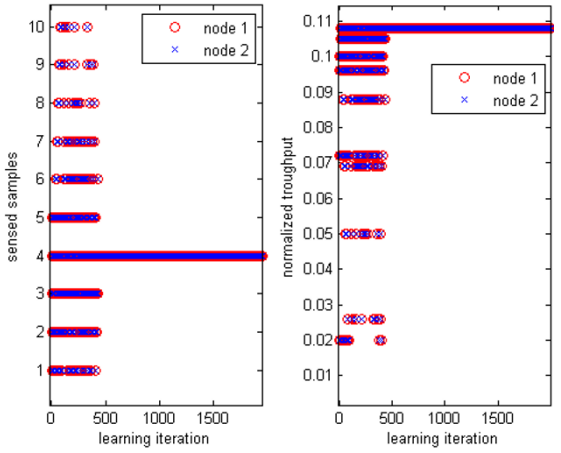

Figure 5 Result of the Q-learning algorithm when no exploration strategy is used ($\epsilon = 0$).

average throughputs are $\hat{R}_{1,\text{opt}}/C_{H_0,1} = 0.096$ and $\hat{R}_{2,\text{opt}}/C_{H_0,2} = 0.144$ and lead to an inferior global cost in Equation (11).

Figure 6 gives the result of the Q-learning algorithms when the exploration strategy described at the beginning of this Section is used. It is observed that the algorithm converges to the optimal solution defined in the previous paragraph.

Table 1 compares the performance of the sensing time allocation algorithm implementation based on the cooperative cost function defined by Equation (23) with the one based on the competitive cost function defined by Equation (20). The cooperative cost function penalizes the actions that lead to a higher than required throughput and as a result performs better (i.e., gives higher realized average throughputs $\hat{R}_j$) than the competitive

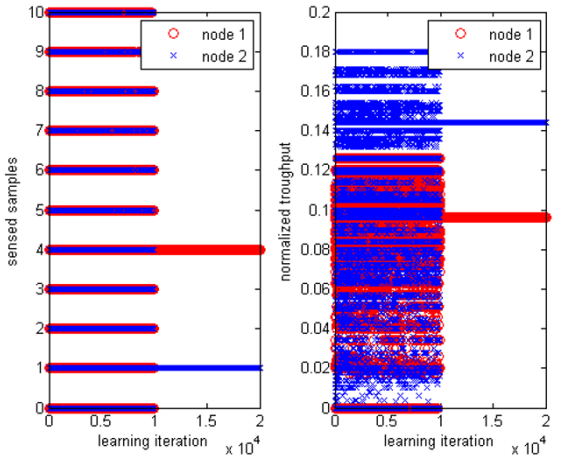

Figure 6 Result of the Q-learning algorithm when an exploration strategy is used ($\epsilon \neq 0$).

cost function, in different scenarios. In particular, it helps achieve fairness among the nodes when one of the nodes has a lower sensing SNR (in which case the other nodes tend to contribute more to the sensing) or when one of the nodes has an inferior channel capacity (in which case this node tends to contribute less to the sensing). The data in Table 1 are the averages of the sensing window lengths and realized throughputs obtained in each scenario.

Figure 7 shows the average normalized throughput that is obtained with the algorithm with respect to parameter $r = T_L/T_H$ when the total duration of execution of the algorithm, equal to $rK\ T_H$, is kept constant. When r decreases, then the learning algorithm is executed more often but the ratio $\frac{n_{H_0,t}}{r}$ becomes a less accurate approximation of $\pi_{H_0}(1 - P_F)p_{00}$ and as a result, the agent becomes less aware of its impact on the false alarm probability. Therefore, there is a tradeoff value for r around $r \approx 10$ as illustrated on Figure 7.

After convergence of the algorithm, if the value of the local SNR γ_1 decreases from 0 dB to -10 dB, the algorithm requires an average of 1200 iterations before converging to the new optimal solution $M_1 = M_2 = 1$. According to Equation (17), each Q-learning iteration requires four additions and five multiplications per node. This result can be compared with the complexity of the centralized allocation algorithm which must be solved numerically. By using a constant step gradient descent optimization algorithm to solve the centralized allocation problem, it was measured that the convergence occurred after an average of four iterations. At each iteration of the algorithm, the partial derivatives of the cost function with respect to the sensing times are evaluated. It can be shown that $18N$ - 1 multiplications and $8N$ - 1 additions are needed for this evaluation. As a result, the centralized allocation algorithm will have a lower computational complexity per node than the Q-learning algorithm. The main advantage of the Q-

Table 1 Average sensing window lengths and realized throughputs obtained with the competitive and cooperative cost functions

		$C_{H_0,1} = 0.6$ $C_{H_0,2} = 0.6$ γ_1 = -5 **dB** γ_2 = -5 **dB**		$C_{H_0,1} = 0.6$ $C_{H_0,2} = 0.6$ γ_1 = 0 **dB** γ_2 = -10**dB**		$C_{H_0,1} = 1.0$ $C_{H_0,2} = 0.2$ γ_1 = -5 **dB** γ_2 = -5 **dB**	
Competitive							
M_1	M_2	2.3	2.0	3.3	0.67	2.5	2
$\hat{R}_1$	$\hat{R}_2$	0.0378	0.0397	0.0556	0.0780	0.0635	0.0133
Cooperative							
M_1	M_2	3.8	3.7	3.7	1.8	3.3	2.3
$\hat{R}_1$	$\hat{R}_2$	0.0423	0.0432	0.0602	0.0779	0.0640	0.0147

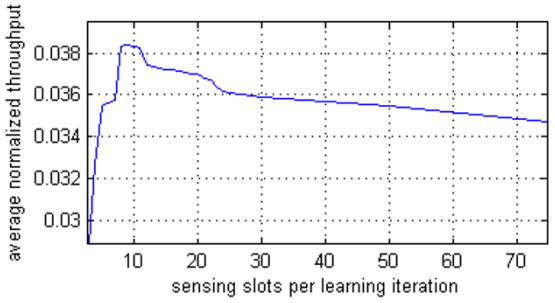

Figure 7 Average throughput achieved by the Q-learning algorithm with respect to parameter $r = \frac{T_L}{T_H}$.

learning algorithm is therefore the minimization of control information sent between the secondary nodes and the coordinator node.

5.2. Power allocation algorithm

The performance of the Q-learning algorithm presented in Section 4 is evaluated by comparison with the optimal centralized power allocation scheme in which a base station having a perfect knowledge of the environment chooses the optimal transmission powers each time there is a change in the environment (i.e., whenever a TDMA time slot ends in any of the L cells). The optimal allocated powers are determined by selecting the transmission powers $(P_1, \ldots, P_L) \in \Psi^L$ that maximize Equation (13) under the constraints given in Equation (12).

The learning algorithm performance metrics considered here is the distance d_t (in dB) between the secondary SINRs obtained with the multi-agent Q-learning algorithm and the secondary SINRs given by the optimal allocation algorithm:

$$d_t = \sqrt{\sum_{l=1}^{N} \left(\mathrm{SINR}_{t,l}^{s} - \widehat{\mathrm{SINR}}_{t,l}^{s} \right)^2} \quad t \in \{1, \ldots, K\} \qquad (34)$$

where $\mathrm{SINR}_{t,l}^{s}$ denotes the secondary SINR measured at iteration t at BS_l in the distributed learning scenario and $\widehat{\mathrm{SINR}}_{t,l}^{s}$ denotes the secondary SINR measured at iteration t at BS_l in the optimal centralized scenario.

The performance is evaluated for $L = 2$ secondary cells with a ray $r_s = 15$ km. The received power from the primary emitter on the protection contour is $P^p = 0$ dBm. Both the primary and the secondary network use a frequency $f_c = 2.45$ GHz. The minimum acceptable primary SINR on the protection contour is $\mathrm{SINR}_{\mathrm{Th}}^{p} = 20$ dB. The desired secondary SINR at the base stations is $\mathrm{SINR}_{\mathrm{Th}}^{s} = 3$ dB. The secondary users are allocated powers ranging from $P_{\min} = 0$ dBm to $P_{\max} = \frac{1}{h_{I_l}^{\mathrm{SU}_l}} \left(-\sigma^2 + \frac{P_p}{\mathrm{SINR}_{\mathrm{Th}}^{p}} \right) = 66.4$ dBm.

The secondary transmission powers $P_{l,\,t}$ are quantized on $\varphi = 15$ levels and the local coordinates $x_{l,\,t}$ of the secondary users are quantized on $\xi = 10$ levels. The Q-learning algorithm is implemented with a learning rate $\alpha = 0.5$, a discount rate $\gamma = 0.9$ and an average randomness for exploration $\bar{\epsilon} = 0.1$.

Figure 8 compares the performance of the power allocation algorithm implementation based on the cooperative cost function defined by Equation (29) with the one based on the competitive cost function defined by Equation (28). The cooperative cost function penalizes the actions that lead to a higher than required secondary SINR and as a result performs better (i.e., gives a lower distance d_t to the optimal solution) than the competitive cost function.

Figure 9 compares the convergence speed of the Q-learning algorithms when different learning frequencies f are used. The Q-learning algorithm converges faster when f increases but the improvement is negligible when $f > 50$. After about 20000 TDMA time slots, the performance of the algorithm is constant and does not depend on the learning frequency.

Figure 10 compares performance of the Q-learning algorithms when different exploration policies are used. The linearly decreasing ϵ strategy defined by Equation (32) converges more slowly than the two other analyzed strategies but leads to better final results. The average d_t of this strategy, computed on the last 50,000 time slots, is equal to 17.5 dB. The full exploration/full exploitation alternance strategy defined by Equation (33) is the strategy that gives the best initial performance but leads to final results that are inferior to those obtained with the linearly decreasing ϵ. The average d_t, computed on the

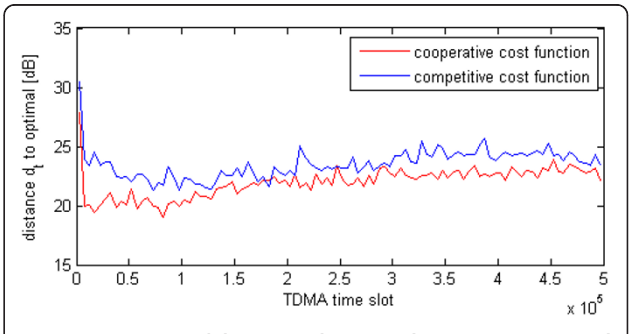

Figure 8 Distance d_t between the secondary SINRs generated by the Q-learning algorithm and the optimal secondary SINRs when using different cost functions in the Q-learning implementation. The randomness of exploration ϵ is constant and the frequency of the learning algorithm $f = 100$.

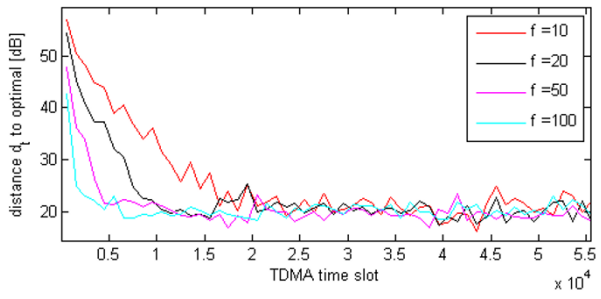

Figure 9 Distance d_t between the secondary SINRs generated by the Q-learning algorithm and the optimal secondary SINRs when using different frequencies of learning *f* in the Q-learning implementation. The randomness of exploration ϵ is constant and a cooperative cost function is used.

last 50,000 time slots, is equal to 18.7 dB. The performance of the constant ϵ strategy defined by Equation (31) is always inferior to the performance of the alternance strategy. The average d_t, computed on the last 50,000 time slots, is equal to 23.0 dB.

The complexity of the decentralized power allocation Q-learning algorithm can be compared to a reference gradient descent centralized power allocation algorithm, similarly to the analysis performed in Section 1. The conclusion is the same as for the sensing time allocation algorithm: the centralized allocation algorithm has a lower computational complexity than the decentralized Q-learning algorithm whose main advantage is therefore that the base stations do not need to exchange control information.

6. Conclusion

In this article, we have proposed two decentralized Q-learning algorithms. The first one was used to solve the problem of the allocation of the sensing durations in a cooperative cognitive network in a way that maximize the throughputs of the cognitive radios. The second one was used to solve the problem of power allocation in a secondary network made up of several independent cells, given strict limit for the allowed aggregated interference on the primary network. Compared to a centralized allocation system, a decentralized allocation system is more robust, scalable, maintainable and computationally efficient.

Numerical results have demonstrated the need for an exploration strategy for the convergence of the sensing time allocation algorithm. It has also been observed that the strategy of keeping the exploration parameter constant in the power allocation algorithm is less efficient than using a linearly decreasing parameter or implementing an alternance between full exploration and full exploitation, this latest exploration policy leading to the fastest convergence of the power allocation algorithm.

It has furthermore been shown that the implementation of a cost function that penalizes the actions leading to a higher than required throughput in the sensing time allocation algorithm gives better results than the implementation of a cost function without such penalty. Similarly, the implementation of a cost function that penalizes the actions leading to a higher than required secondary SINR in the power allocation algorithm gives better results than the implementation of a cost function without such penalty.

Finally, it has been shown that there is an optimal tradeoff value for the frequency of execution of the sensing time allocation algorithm. The power allocation algorithm has been shown to converge faster when its frequency of execution increases, until the frequency reaches an upper bound where the increase of the convergence speed gets insignificant.

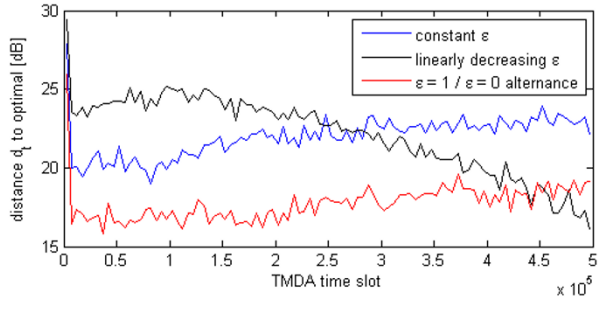

Figure 10 Distance d_t between the secondary SINRs generated by the Q-learning algorithm and the optimal secondary SINRs when using different exploration strategies in the Q-learning implementation. The learning frequency is $f = 100$ and a cooperative cost function is used.

Competing interests
The authors declare that they have no competing interests.

Received: 20 May 2011 Accepted: 10 April 2012 Published: 10 April 2012

References

1. FK Jondral, TA Weiss, Spectrum pooling: An innovative strategy for the enhancement of spectrum efficiency. IEEE Radio Commun. **42**(3), S8–S14 (2004)
2. B Aazhang, A Sendonaris, E Erkip, User cooperation diversity. Part I: system description IEEE Trans Commun. **51**(11), 1927–1938 (2003)
3. GB Bazerque, JA Giannakis, Distributed spectrum sensing for cognitive radio networks by exploiting sparsity. IEEE Trans Signal Process. **58**(3), 1847–1862 (2010)
4. E Peh, Y-C Liang, Y Zeng, AT Hoang, Sensing-throughput tradeoff for cognitive radio networks, IEEE Trans. Wirel Commun. **4**(7), 1326–1337 (2008)
5. S Stotas, A Nallanathan, Sensing time and power allocation optimization in wideband cognitive radio networks, in *GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, Miami, pp. 1–5 (2010)
6. W Beibei, KJR Liu, TC Clancy, Evolutionary cooperative spectrum sensing game: how to collaborate? IEEE Trans. Commun. **58**(3), 890–900 (2010)
7. S Shankar, C Cordeiro, Analysis of aggregated interference at DTV receivers in TV bands, in *Proceedings of the 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, Singapore, pp. 1–6 (2008)

8. R Tandra, SJ Shellhammer, S Shankar, J Tomcik, Performance of power detector sensors of DTV signals in IEEE 802.22 wrans, in *Proceedings of First International Workshop on Technology and Policy for Accessing Spectrum*, Boston, (2006)
9. A Dejonghe, A Bahai, LV der Perre, M Timmers, S Pollin, F Catthoor, Accumulative interference modeling for cognitive radios with distributed channel access, in *Proceedings of the 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, Singapore, pp. 1–7 (2008)
10. A Galindo-Serrano, L Giupponi, Distributed q-learning for aggregated interference control in cognitive radio networks. IEEE Trans Veh Technol. **59**, 1823–1834 (2010)
11. P Liviu, L Sean, Cooperative multi-agent learning: the state of the art, Auton. Agents Multi-Agent Syst. **11**(3), 387–434 (2005)
12. H Li, Multi-agent Q-learning for competitive spectrum access in cognitive radio systems, in *5th IEEE Workshop on Networking Technologies for Software Defined Radio Networks*, pp. 1–6 (2010)
13. H Urkowitz, Energy detection of unknown deterministic signals, in *Proceedings of the IEEE*. **vol 55**, 523–531 (1967)
14. FF Digham, M-S Alouini, MK Simon, On the energy detection of unknown signals over fading channels. IEEE Trans Commun. **55**(1), 21–24 (2007)
15. J Ma, G Zhao, Y Li, Soft combination and detection for cooperative spectrum sensing in cognitive radio networks. IEEE Trans Wirel Commun. **7**(11), 4502–4507 (2008)
16. Y-C Liang, Y Zeng, ECY Peh, AT Hoang, Sensing-throughput tradeoff for cognitive radio networks. IEEE Trans Wirel Commun. **7**(4), 1326–1337 (2008)
17. I Millington, *Artificial Intelligence for Games*, (Morgan Kaufmann Publishers, San Fransisco, CA, 2006), pp. 612–628
18. C Watkins, P Dayan, Technical note: Q-learning. Mach Learn. **8**, 279–292 (1992). doi:10.1023/A:1022676722315
19. C Wu, K Chowdhury, M Di Felice, W Meleis, Spectrum management of cognitive radio using multi-agent reinforcement learning, in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Industry track, AAMAS'10, (International Foundation for Autonomous Agents and Multiagent Systems)*, Richland, SC, pp. 1705–1712 (2010)

doi:10.1186/1687-1499-2012-138
Cite this article as: van den Biggelaar *et al.*: **Sensing time and power allocation for cognitive radios using distributed Q-learning.** *EURASIP Journal on Wireless Communications and Networking* 2012 **2012**:138.