International Journal of Future Generation Communication and Networking
Vol.7, No.1 (2014), pp.29-42
http://dx.doi.org/10.14257/ijfgcn.2014.7.1.03

Admission Control Using Stochastic Learning Games in Cognitive Radio Networks

Mohammed Raiss-El-Fenni[1] and Mohamed El Kamili[2]

[1]HECI, 30000, Fez, Morocco
[2]LiM, USMBA 30000, Fez, Morocco
Raiss.med@gmail.com, mohamed.elkamili@usmba.ac.ma

Abstract

A cognitive radio wireless network is an emerging communication paradigm to effectively address spectrum scarcity challenge. In this paper, we investigate the performance improvement gained by applying cognitive radio to a multiple Wireless Service Providers (WSPs). We consider several independent WSPs and two types of users: primary (licensed) and secondary (unlicensed) users. We have proposed a scheme for secondary users to manage their handoff based on a threshold on the number of channels fixed by the WSP. As each WSP evolves in a random environment, we propose a game theoretic framework to find a Nash equilibrium and a stochastic learning algorithm to converge to this equilibrium. A Markov process, with continuous time and finite state space, models the system. Numerical results show the system equilibrium points and the conditions to converge to the best one of them to increase the spectrum utilization of cognitive users.

Keywords: *Cognitive radio, Admission Control, Migration rate, Nash Equilibrium, Stochastic learning, LRI Algorithm*

1. Introduction

Cognitive radio is a paradigm for wireless communication in which either a WSP or a wireless node changes its transmission or reception parameters to communicate efficiently avoiding interference with licensed or unlicensed users [1]. This alteration of parameters is based on the active monitoring of several factors in the external and internal radio environment, such as radio frequency spectrum, user behavior and network state. In a network supporting cognitive radio, secondary users (SUs) opportunistically use the spectrum that is normally assigned to primary users (PUs) but not being used at a particular time and geographic location. Thus, a secondary user must leave the channel once detecting the presence of a licensed user. At the same time, the SU will scan all other channels and handoff to another unused one if available; otherwise, its communication is interrupted. Once interrupted, the SU can either leave the system or wait in a queue so that its connection is terminated or suspended. In [2], the authors propose a new dynamic spectrum access scheme for cognitive radio wireless networks with and without buffering for secondary users to avoid direct blocking. A Markov model was developed to analyze the proposed spectrum sharing policies. The result indicates that the buffer is able to significantly reduce the SU blocking probability and no completion probability with very minor increased forced termination probability. A rigorous study of performance in opportunistic spectrum access systems that limit disruptions to unpredictable primary users is presented in [3]. The goal was to understand whether dynamic spectrum access could provide reliable spectrum to secondary users while respecting hard disruption limits that protect primary-user transmissions. Virtual unlicensed spectrum is another way to share bands with primary users without interferences. Thus, in [4] a cognitive radio approach for usage of virtual unlicensed spectrum is presented. A vision of a cognitive radio based approach that uses allocated spectrum in an opportunistic manner to create virtual unlicensed bands i.e., bands that are shared with the primary users on a non-interfering basis. Dynamic spectrum management techniques are used to adapt to immediate local spectrum availability. The authors define the system requirements for this approach, as well as the general architecture and basic physical and link layer functions.

In general, there are two basic operation models for SUs: OSA that is also called the spectrum overlay paradigm versus spectrum sharing (SS) or the spectrum underlay paradigm. In the OSA model, the SUs are allowed to transmit over the band of interest when all the Pus is not transmitting at this band. One essential enabling technique for OSA based SUs is spectrum sensing; where the SUs individually or collaboratively detect active PU transmissions over the band, and decide to transmit if the sensing results indicate that all the PU transmitters are inactive at this band with a high probability. The key operation is spectrum sensing; the interested readers may refer to, *e.g.*, [5, 1] for an overview of the state of art results in this area. As a counterpart, the SS model allows the SUs to transmit simultaneously with PUs at the same band even if they are active, provided that the SUs know how to control their resultant interference [6]. For instance, in [7] packet collision probability was considered as the PU protection requirement. Under this requirement, the SU must guarantee that the packet collision probability of a PU packet is less than a certain threshold specified by the PU.

As SUs are required to share the interference channel information and power strategies to conduct the game with pricing mechanisms that incur the frequent exchange of information, the authors in [8] propose a distributed power control algorithm for CR networks through reinforcement learning, and provide an incomplete-information power control game in CR networks.

In this paper we consider a system with multiple Wireless Service Providers (WSPs) where multiple SUs are allowed to access the unused licensed spectrum bands without conflicting with PUs. In order to protect the QoS of PUs, each WSP reserves a number of channels allowed to SUs by fixing a threshold. We assume that SUs are equipped with cognitive radio equipment and are able to detect the presence of the licensed users. How a service provider should choose the threshold is a crucial task. Indeed, decreasing the threshold could produce a high migration rate of SUs to other WSPs and brings down their performance, while increasing it could degrades the QoS offered to PUs. In practice, there is no reserved bandwidth to SUs, and the threshold defined can be interpreted as reserving all available channels to PUs. Therefore, SUs use only empty channels when the PUs is absent or inactive. Our paper covers this case. The interest of this work is to study the benefit of including such threshold by WSPs in term of performance while insuring a QoS to PUs all the time. Indeed, a user can use licensed channels his real time traffic in order to guarantee the desired QoS, while the best effort one can be performed using the unlicensed channels with cognitive equipment. Another reason of such framework can be reducing the total cost of service. We formulate the problem as a non-cooperative game between WSPs, and we look for the equilibrium strategy that allows achieving high spectrum efficiency while conforming to the interference constraint of PUs.

2. System Description

The system under consideration is a collection of N WSPs with infrastructure deployed in the same geographical area and covers a shared pool of end users (see Figure 1 with 3 service providers). The WSP i possesses K_i channels and without loss of generality each user requires the capacity of one channel for the duration of its service in order to fulfill its requirements.

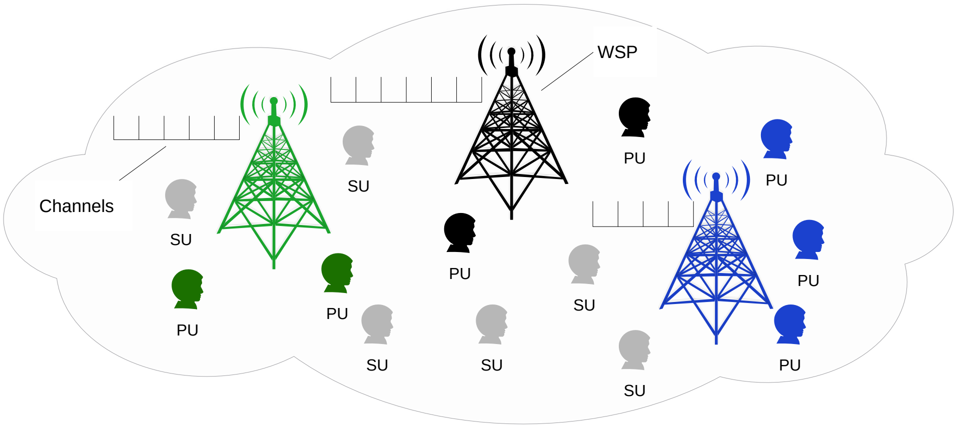


Figure 1. Illustrative Example of the System with Three WSPs

We consider two types of users:

- The PUs related to a specific WSP with a guaranteed QoS, dedicated channels and without the possibility to have service in other WSPs.
- The SUs who are not attached to any WSP and sense all channels of the system looking for spectrum holes.

The system considered has a total capacity of $M = \sum_{i=1}^{N} K_i$. However, the WSPs belonging to the system are completely independent of each other, with primary users only being served by their home WSP. As a result, when a WSP reaches its capacity, it must start blocking calls, as it has no available channels. For SUs, if resources are not available in a WSP, the SU can switch to another WSP j ($j \neq i$) with a uniform probability on all networks with free resources. As a SU may be pushed out if it uses a licensed channel and no other resource is available for PUs, an occupancy threshold Th for SUs has been included in the model. However, our model still cover the classical scenario where PUs have the right to use all channels, and SUs take advantage of spectrum only when PUs are absent. The threshold was included in order to study how our scenario may improve the performance of SUs under the QoS constraint required by PUs. SUs can occupy channels until the threshold. Once exceeded, the new calls are blocked. As for PUs, they can use in addition of the rest of channels, the empty reserved resources for SUs. A PU didn't interrupt the service of a SU under any condition. Indeed, a PU is blocked if $n_{PU}^{i} = K_i$, and a SU is blocked if $n_{SU}^{i} \geq Th_i$, where n_{PU}^{i} (resp. n_{SU}^{i}) is the number of Pus (resp. SUs) in the WSP i, and Th_i is its corresponding threshold.

3. Admission Control

a. Transition Rates and System State

In order to analyze the behavior of the WSPs and the interactions between them in the sense of SUs migrations, the state of the system at time t is defined by

$$\vec{n}(t) \triangleq \left(n_{PU}^{1}(t),\ldots,n_{PU}^{N}(t),n_{SU}^{1}(t),\ldots,n_{SU}^{N}(t)\right).$$

Then, we model the process $\{\vec{n}(t),\ t > 0\}$ as 2N-dimensions quasi-birth and death Markov chain with continuous time and state space. Thus, we define the space of admissible states as

$$E = \{\vec{n} \in \mathbb{N}^{2N} \mid n_{PU}^{i} + n_{SU}^{i} \leq K_{i} \text{ and } n_{SU}^{i} \leq Th_{i}; i = 1,...,N\}.$$

A perfect channel is assumed in which a channel is either busy or idle. It is also assumed that radio systems always detect radio resource an allocation of each other's, here the radio resource refers to bandwidth. We assume that spectral scanning is performed instantaneously, so there is no scanning delay. We assume that PUs arrivals to WSP i, follow a Poisson distribution with rate $/_{i}^{PU}$ and their service time is exponentially distributed with mean $1/ m_{PU}^{i}$. We assume also that SUs service time to WSP i is distributed exponentially with$1/ m_{SU}^{i}$. However, the arrival rate of SUs depends on the global state ($\vec{n} \in E$) of the system. The Poisson process has been found to provide a good representation for arrival calls and service time of a call while Internet traffic exhibits a correlation structure over a wide span of time scales.

Hence, the arrival rate of a SU to WSP i is given by

$$\lambda_{SU}^{i}(\vec{n}) = \lambda_{new,SU}^{i} + \sum_{j \neq i} \lambda_{SU}^{j,i}, \qquad (1.1)$$

$$\text{If } n_{SU}^{i} < Th_{i} \text{ and } n_{PU}^{i} + n_{SU}^{i} < K_{i}$$

where $/_{new,SU}^{i}$ is the arrival rate of new SUs to WSP i, and $/_{SU}^{j,i}$ is the migration rate of SUs from WSP j to i ($j^{1}\ i$).

We recall that the WSPs are totally independent. The migration of SUs is the only interaction between them. According to a previous work [9], the computation of the steady state can be approximated by considering the steady state of each WSP. Let us define the sub state space of WSP i as follows

$$E_{i} = \{\overrightarrow{n_{i}} \in \mathbb{N}^{2} \mid n(i) \leq K_{i} \text{and } n_{SU}^{i} \leq th_{i}; i = 1,\ldots,N\},$$

where $\overrightarrow{n_{i}} = (n_{PU}^{i}, n_{SU}^{i})$ and $n(i) = n_{PU}^{i} + n_{SU}^{i}$. The steady state probability of WSP i is defined by $\pi_{i} = \{\pi_{i}(\overrightarrow{n_{i}}) \mid \overrightarrow{n_{i}} \in E_{i}\}$.

This sub steady state is given by using the standard argument for queuing networks. Hence, based on the BCMP theorem [10] for multiple classes, the state probability in the closed product-form is given with $\rho_{c}^{i} = \dfrac{\lambda_{c}^{i}(\vec{n})}{\mu_{c}^{i}}$ ($c = PU, SU$) as follows:

$$\pi_i(\overrightarrow{n_i}) = \frac{1}{G_i}\frac{(\rho_{PU}^i)^{n_{PU}^i}}{n_{PU}^i\,!}\frac{(\rho_{SU}^i)^{n_{SU}^i}}{n_{SU}^i\,!}, \qquad (1.2)$$

where G_i is the normalizing constant over the sub space E_i and its given by:

$$G_i = \sum_{\overrightarrow{n_i}\in E_i}\frac{(\rho_{PU}^i)^{n_{PU}^i}}{n_{PU}^i\,!}\frac{(\rho_{SU}^i)^{n_{SU}^i}}{n_{SU}^i\,!}.$$

The arrival rate $/_{SU}^{j,i}$ (used in (1.1)) is unknown and depends on the random number of SUs of WSP j that go elsewhere. These rates are calculated iteratively in the next sub-section.

b. Migration Rates

A SU from WSP j goes elsewhere if $n_{SU}^j = Th_j$ (i.e. SUs use all reserved resources of j). Let $a(\vec{n})$ be the number of WSPs that have unused resources at state $\vec{n}$, then we have

$$a(\vec{n}) = \sum_{i=1}^{N}\mathbf{1}_{\{n(i)<K_i,\, n_{SU}^i<Th_i\}}.$$

Let $\pi(\vec{n})$ be the probability that the system is in state $\vec{n}$ and is defined as follows:

$$\pi(\vec{n}) = \frac{1}{G}\prod_{i=1}^{N}\pi_i(\overrightarrow{n_i}), \qquad (1.3)$$

where G is a constant given by $G = \sum_{\vec{n}\in E}\prod_{i=1}^{N}\pi_i(\overrightarrow{n_i})$. Then, the arrival rate of a SU from WSP j to i is given by

$$\lambda_{SU}^{j,i} = \sum_{\substack{\vec{n}/n_{SU}^j=Th_j \\ n(i)<K_i,\, n_{SU}^i<Th_i}}\frac{\pi(\vec{n})}{a(\vec{n})}\lambda_{new,SU}^j. \qquad (1.4)$$

In order to obtain the arrival rate $\lambda_{SU}^i(\vec{n})$ from (1.1), we first need to compute the migration rate of SUs ($/_{SU}^{j,i}, j^{1}\ i$). To this aim, we follow the Algorithm 1:

Algorithm 1: Migration rate computing

- Initialize the migration rates by zero in (1.1): $/_{old,SU}^{j,i} = 0$.
- Calculate the steady-state probability $\pi(\vec{n})$ from (1.3).
- Derive the new values of migration rates from (1.4) and denote it by $/_{SU,new}^{j,i}$.
- Check the convergence of the migration rates between old and new rates, *i.e.*, if $|/_{SU,new}^{j,i} - /_{SU,old}^{j,i}| < X$, where X is a very small positive number, then the new migration rates will be used to compute the performance metrics. Otherwise, go to step 2 with the new migration rates as initial values, then the iterations are continued until to reach the convergence of rates.

Note that, since the Markov chain has a stationary distribution π, a migration rate $\lambda^{j,i}_{SU}$ exists and can be estimated by the previous Algorithm.

c. Performance Evaluation

We have used some classic QoS metrics to evaluate the performance of the system: blocking probability and cognitive throughput. Each is calculated based on the state and transition probabilities as follows:

i. Blocking Probability

A new PU to WSP i is blocked if all resources of this WSP are used.

$$P_B(i,PU)=\sum_{\vec{n}\in E}\pi(\vec{n}).\mathbf{1}_{\{n^i_{PU}+n^i_{SU}+1>K_i\}}.$$

A new SU is refused to have service in WSP i if all resources of this WSP are used or the threshold is reached by SUs. Then the blocking probability of a SU is

$$P_B(i,SU)=\sum_{\vec{n}\in E}\pi(\vec{n}).(\mathbf{1}_{\{n^i_{PU}+n^i_{SU}+1>K_i\}}+\mathbf{1}_{\{n^i_{SU}+1>Th_i\}}),$$

Then, the overall blocking probability for a new SU flows to the all system is given by

$$P^B_{SU}=\frac{\sum_{i=1}^{N}\lambda^i_{new,SU}P_B(i,SU)}{\sum_{i=1}^{N}\lambda^i_{new,SU}}.$$

ii. Spectrum Utilization

We define the spectrum or bandwidth utilization as follows:

$$Uti=\frac{1}{M}\sum_{i=1}^{N}\sum_{\vec{n_i}\in E_i}\pi(\overrightarrow{n_i})(n^i_{PU}+n^i_{SU}).$$

Since SUs have no guarantee of their QoS, our aim in this work is to maximize their throughput in each WSP i by taking into consideration the quality of service reserved to PUs. We recall that service providers are completely independent. No information about other WSPs resource allocation is available. This kind of problems in a random environment can be solved using no cooperative games to find a Nash equilibrium and Stochastic learning algorithms to converge to this equilibrium.

4. Stochastic Automata Games

In this section, we briefly introduce the concept of automata games [11]. The process of a learning automaton proceeds as follows. At each time, the automaton interacts with the environment by choosing randomly from a finite set of actions based on some probability distribution (initially, each action can be selected with equal probability). For every action that it chooses, the environment evaluates that action and sent a feedback to the automaton based on which the next action is chosen and the procedure is repeated. An important property of the learning automaton is its ability to improve its performance with time in an unknown environment.

a. Game Formulation

In a game, instead of one automaton playing with the environment, N automatons take part. Each automaton is described by a quadruple $\{A_i, r_i, L_i, p_i\}$, where A_i is a finite set of actions or pure strategies with a_i the cardinality of A_i. The result of each play of the game consists of each automaton independently choosing an action according to their action probabilities vectors. The reaction to a play is a random payoff to each player denoted by r_i. The action probability distribution of the i^{th} automaton player is denoted by $p_i(k) = (p_{i1}(k), p_{i2}(k), \cdots, p_{ia_i}(k))$. Where $p_{ij}(k)$ is the probability that the i^{th} automaton player chooses the j^{th} pure strategy at instant k. $p_i(0)$ is the initial mixed strategy of player i. In the absence of extra information about strategies, each player can assign equal probabilities to all actions. The elements of p_i are updated at each time k with $p_i(k+1) = L_i(p_i(k), a_i(k), r_i(k))$ where L_i is the stochastic learning algorithm.

Each WSP can improve the cognitive throughput by increasing the value of threshold Th at the cost of high cognitive arrival rate to other WSPs, and brings down their throughput. In Figure 2, when WSP 2 increases its threshold, the migration rate to WSP 1 of SUs increases, and thus the arrival rate of SUs increases also. As a consequence, the cognitive throughput decreases.

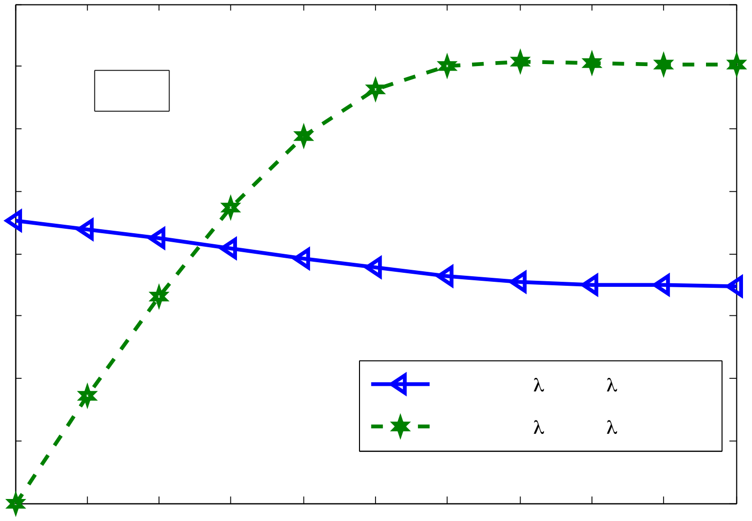


Figure 2. Cognitive Throughput with Two WSPs

In order to keep a WSP from a selfish behavior in this no cooperative game, the steady state is computed and known as Nash Equilibrium. Formally the no cooperative game is expressed as:

$$\underset{Th_i}{\text{Maximize}}\, Thp_i^{SU}(Th_i, Th_{-i}) \quad s.t P_B(i,PU) £ P_t, i = 1,...,N$$

where P_t is the maximal tolerated blocking probability of PUs in order to maintain a desired QoS, and Th_{-i} is the strategy vector of other players, and it's defined by $Th_{-i} = (Th_1,...,Th_{i-1},Th_{i+1},...,Th_N)$. The solution of this game is given in the sense of NE [12] as follows.

Definition 1: *A thresholds vector Th=(*$Th_1,...,Th_N$*) is a NE of the game if* $Thp_i^{SU}(Th_i, Th_{-i}) \geq Thp_i^{SU}(\widetilde{Th_i}, Th_{-i})$ *where* $i = 1,...,N$, $\widetilde{Th_i} \in Th$.

Once reached the NE, no player can improve the value of its utility by unilaterally deviating. The action tuples corresponding to the NE are then predicted as the most likely steady state outcome of the game.

b. Discrete Stochastic Learning Game

The learning algorithm used by each player is given in Algorithm 2. This Algorithm is known as Linear Reward-Inaction (L_{R-I}) Algorithm [13]. We can see from the rule of updating action probabilities (step 4 of the algorithm) that this algorithm is completely decentralized which is consistent with our model, and also we can see that $\{P(k), k ³ 1\}$ is a Markov process.

Algorithm 2:

- Initialize $p_i(k) = 1/a$, for $1 £ i £ a$.
- Choosing randomly a threshold Th_i according to probability vector p_i.
- Each player computes its utility function Thp_i^{SU} using the chosen threshold and obtains a payoff based on the set of all actions.
- Each player updates its action probability p_i according to the rule:
  $p_i(k+1) = p_i(k) + D.Thp_i^{SU}(1 - p_i(k))$
  $p_j(k+1) = p_j(k) - D.Thp_i^{SU} p_j(k) \quad for \quad all \quad j \neq i.$
  Where D is the smallest step size.
- Check the convergence between the old probability value $p_i(k-1)$ and the new one $p_i(k)$ *,i.e.,if* $|p_i(k) - p_i(k-1)| < e$, where e is a very small positive number, then the algorithm has converged. Otherwise, go to step 3 with the new probability vector $p_i(k)$ as initial values, and the iterations are continued until to reach the convergence.

c. Theoretical Analysis of the Learning Algorithm

A general method of analysis for such learning algorithms is the called ODE method. This method is based on the fact that for small values of the learning parameter (step size) D used in (step 4 of the Algorithm 2), the long-term behavior of the algorithm can be approximated by an associated ordinary differential equation (ODE). The learning algorithm can be represented by [14]

$$P(k+1) = P(k) + \mathrm{D}.G(P(k), \mathsf{a}(k), \mathsf{b}(k)),$$

where $\alpha(k) = (\alpha_1(k), \cdots, \alpha_N(k))$ is the vector of actions selected by the N automata and $\beta(k) = (\beta_1(k), \cdots, \beta_N(k))$ the vector of resulting payoffs or utilities. G represents the updating action probabilities specified by the learning algorithm. We define a function g as the following conditional expectation

$$g(P) = E[G(P(t), \mathsf{a}(t), \mathsf{b}(t)) \mid P(k) = P].$$

The evolution of $\{P(k)\}$ is dependent on the step size D. With different values of D, we would be generating different sequences of P(k). Since $\mathrm{D} \to 0$, P(k) will converge weakly to P' which is the solution of the ODE.

$$\frac{dP}{dt} = g(P), P(0) = P_0,$$

We analyze a simple example by considering the case of two WSPs, each one of them can choose between a actions ($Th\,\hat{\text{I}}\ \{1,2,...,a\}$). The following pair of game matrices defines the game

$$\text{WSP1} = \begin{pmatrix} Thp_{11} & Thp_{12} & \cdots & Thp_{1a} \\ Thp_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ Thp_{a1} & \cdots & \cdots & Thp_{aa} \end{pmatrix} \quad \text{WSP2} = \begin{pmatrix} Thp'_{11} & Thp'_{12} & \cdots & Thp'_{1a} \\ Thp'_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ Thp'_{a1} & \cdots & \cdots & Thp'_{aa} \end{pmatrix}$$

If WSP1 chooses action $i\,\hat{\text{I}}\ \{1,2,...,a\}$ and WSP2 chooses action $j\,\hat{\text{I}}\ \{1,2,...,a\}$, the payoff to WSP1 is Thp_{ij} and the payoff to WSP2 is $Thp¢_{ij}$. Let $0 \text{ £ } \mathsf{g}_i \text{ £ } 1$ be the probability of WSP1 choosing action i with $\overset{s}{\underset{i=1}{\text{å}}} \mathsf{g}_i = 1$ and let $0 \text{ £ } \mathsf{d}_i \text{ £ } 1$ be the probability of WSP2 choosing action i with $\overset{s}{\underset{i=1}{\text{å}}} \mathsf{d}_i = 1$. Define d_{1j} (resp d_{2j}) as the reward probability of action a_j of player WSP1 (respectively WSP2). Then, we can express explicitly the differential equation whose solution characterizes the long-term behavior of the algorithm as:

$$f_{1i} = \frac{d\mathsf{g}_i}{dt} = \overset{s}{\underset{j=1}{\text{å}}} \mathsf{g}_i\mathsf{g}_j(d_{1i} - d_{1j})$$

$$= \mathsf{g}_i d_{1i}(\overset{s}{\underset{j=2}{\text{å}}} \mathsf{g}_j) - \mathsf{g}_i(\overset{s}{\underset{j=2}{\text{å}}} \mathsf{g}_j d_{1j})$$

With

$$d_{1i} = E[\beta(k) \mid \alpha(k) = a_i] \quad i = 1, \cdots, s$$

$$= E[\beta_{1i} \mid a_1 = Th_i] = \sum_{j=1}^{s-1} \delta_j Thp_{ij} + (1 - \sum_{j=1}^{s-1} \delta_j) Thp_{is}$$

and

$$f_{2i} = \frac{d\delta_i}{dt} = \sum_{j=1}^{s} \delta_i \delta_j (d_{2i} - d_{2j})$$

$$= \delta_i d_{2i} (\sum_{j=2}^{s} \delta_j) - \delta_i (\sum_{j=2}^{s} \delta_j d_{2j})$$

With

$$d_{2i} = E[\beta(k) \mid \alpha(k) = a_i] \quad i = 1, \cdots, s$$

$$= E[\beta_{2i} \mid a_2 = Th_i] = \sum_{j=1}^{s-1} \gamma_j Thp'_{ji} + (1 - \sum_{j=1}^{s-1} \gamma_j) Thp'_{si}$$

From the above equations, we can solve the system by setting the left side of equations to zero and get the equilibrium solution.

5. Numerical Results

We turn in this section to study a concrete example that will allow us to measure different metric performances of the system. We consider a system consisting of two WSPs. The set of parameters used in this section is summarized in the following table

Parameters	WSP1	WSP2
K	10	10
μ_{PU}	0.3	0.4
μ_{SU}	0.3	0.4
$\lambda_{new,SU}$	0.4	0.3

a. Uniqueness of Equilibrium

In a game, the case of multiple pure Nash equilibrium can occurs. This situation is more present when we have discrete strategies. In most figures we focus on the cognitive throughput as a measurement metric. Figure 3 shows the best response of two WSPs while limiting blocking of PUs to 0.1%. We have two pure equilibriums $Th^* = (6,4)$ that we denote by the first equilibrium where $Th1^* = 6$ and $Th2^* = 4$, and $Th^* = (7,3)$ denoted by the second equilibrium. The throughput received when playing the first equilibrium is the vector (0.678,0.454), while with the second equilibrium the WSPs obtained (0.6619,0.436). It is easy to see that this makes a little difference between the two equilibriums, since the total payoff of the first one is better than the second. If we had no differences between the two equilibriums and both of them were efficient, the problem would be more complicated and the game of this kind will raises in the called coordination games. Questions to which we have to answer are: the algorithm will converge to the best equilibrium? And in which cases the second equilibrium may present an attractor to the algorithm?

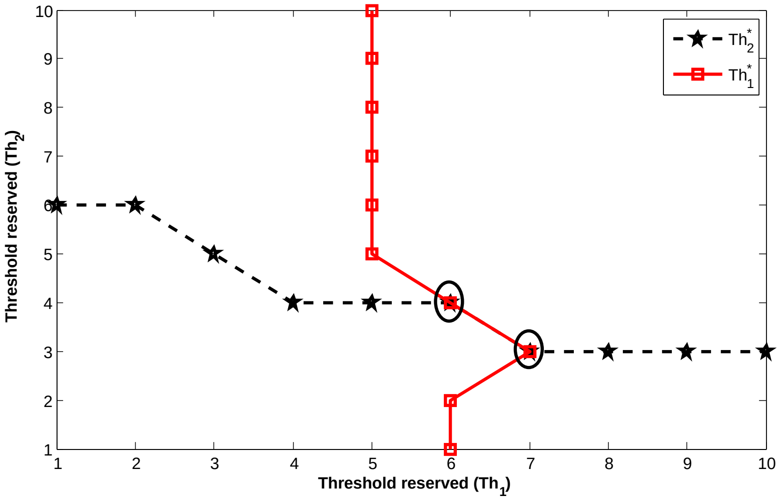


Figure 3. Best Response of Two WSPs

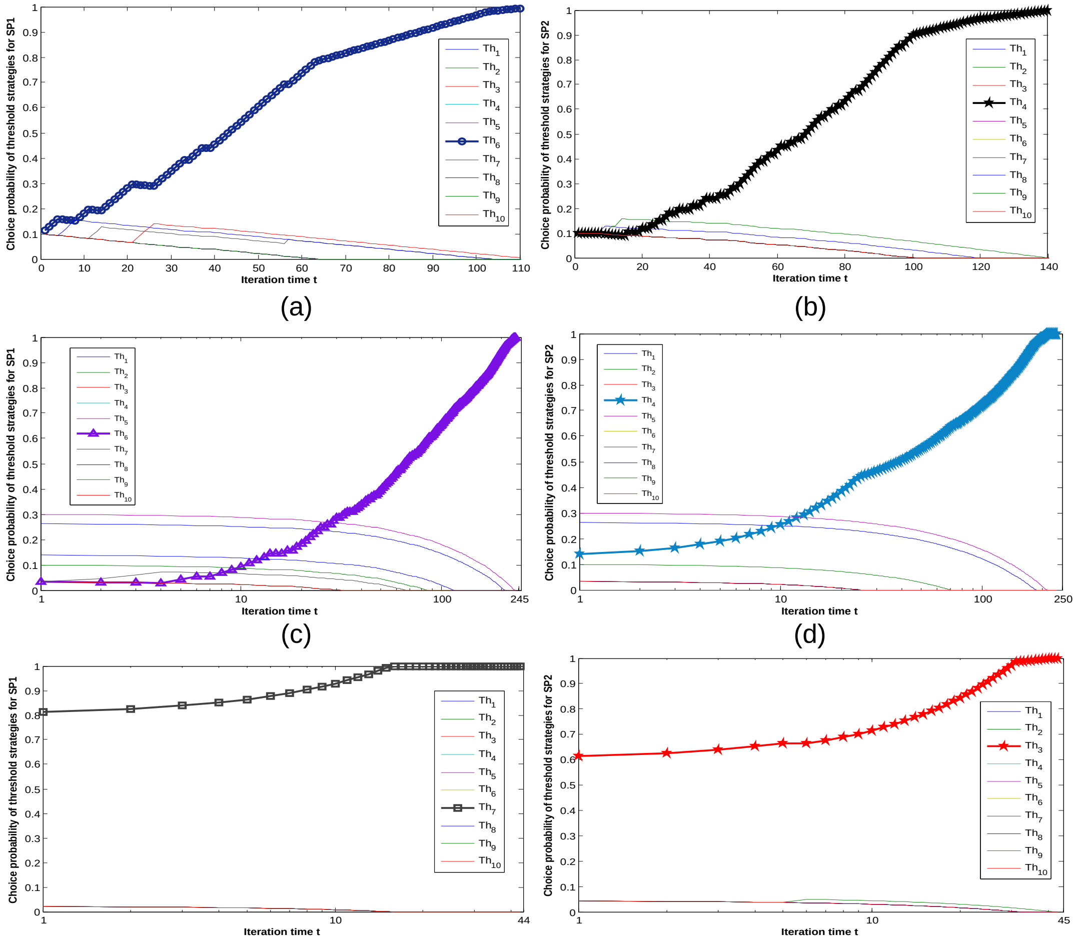
b. Stochastic Learning Game

In Figure 4, we plot the choice probability of the threshold strategies to try to answer the questions asked before. We present three different scenarios, the first one with equally initial probability of actions. The second scenario with a random initial distribution and the last one with a distribution close to an equilibrium point. We choose these scenarios to cover the majority of cases and observe the behavior of the algorithm in each case. With a uniform initial distribution (Figure 4a) and 4(b)), after a small number of iterations, the algorithm starts learning the optimal actions and converges to the best equilibrium $Th = (6,4)$. This speed of learning is decreased when we choose a random vector of action probabilities, this is due the time needed to correct these wrong probabilities, but the convergence remains to the best equilibrium. For illustration, we choose P1(t=0)=[0.2 0.035 0.045 0.035 0.3 0.035 0.035 0.16 0.12 0.035] and P2(t=0)=[0.21 0.035 0.035 0.14 0.3 0.04 0.045 0.045 0.1 0.045] (figure 4(c) and 4(d)). As depicted in figure 3 there is two pure Nash equilibrium, the last scenario shows the case when the convergence is to the second equilibrium. In figure 4(e) and 4(f), we consider the following example P1(t=0)=[0.022 0.022 0.022 0.022 0.022 0.022 0.8 0.022 0.022 0.022] and P2(t=0)=[0.044 0.044 0.6 0.044 0.044 0.044 0.044 0.044 0.044 0.044], which means the case when the starting distribution is too close to the equilibrium point. This is the only case when the algorithm does not converge to the best equilibrium.

We have simulated more than these three scenarios, and all paths of the learning process converge to the optimal strategy independently of their initial probability distribution. The extreme cases are those initial probabilities points too close to the weak equilibrium. But we can always keep those situations from happening by uniformly setting all actions probabilities. Without advance information, this initialization is the most natural and fair thing we can do.

6. Conclusion

In this paper, we have studied the QoS and the performance evaluation in wireless networks. The system proposed is a set of WSPs competing with each other's. The model developed describes the interaction between users in one side, and between users and service providers in the other side. A Markov process, with continuous time and finite state space, models the system. Furthermore, we have seen that improving the service time of SUs is possible with respect to the QoS offered to PUs. In addition, several numerical results are presented in this paper in order to analyze the game equilibrium. The learning algorithm allows to SUs to improve their service time without deteriorating the QoS of PUs.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 5. Choice Probability of Threshold Strategies with PUs' QoS Fixed to 0.1% for Different Initial Distribution Probabilities

References

[1] B. Zhao and S. Shimamoto, "Energy-aware cooperative spectrum sensing with inter-channel interference cancellation for cognitive radio networks", Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on, **(2011)** August, pp. 1-6.

[2] Y. Zhang, "Dynamic spectrum access in cognitive radio wireless networks", Proc. IEEE ICC 2008, Beijing, China, **(2008)** May, pp. 4927-4932.

[3] V. Kone, L. Yang, X. Yang, B. Y. Zhao and H. Zheng, “The effectiveness of opportunistic spectrum access: A measurement study”, Networking, IEEE/ACM Transactions on, vol. 1, **(2012)**, pp. 99.

[4] D. Cabric, S. M. Mishra, R. W. Brodersen, A. Wolisz and D. Willkomm, “White paper: Corvus: A cognitive radio approach for usage of virtual unlicensed spectrum”, available: http://bwrc.eecs.berkeley.edu/research/mcma/cr white paper final1.pdf.

[5] Y. H. Zeng, Y.-C. Liang, A. T. Hoang and R. Zhang, “A review on spectrum sensing for cognitive radio: challenges and solutions”, EURASIP J. Advances Signal Process, **(2010)**.

[6] H. B. Salameh, M. Krunz and D. Manzi, “An efficient guard-band-aware multi-channel spectrum sharing mechanism for dynamic access networks”, Global Telecommunica- tions Conference (GLOBECOM 2011), IEEE, **(2011)** December, pp. 1-5.

[7] E. Jung and X. Liu, “Opportunistic spectrum access in multiple-primary-user environments under the packet collision constraint”, Networking, IEEE/ACM Transactions on, vol. 20, no. 2, **(2012)** April, pp. 501-514.

[8] P. Zhou, Y. Chang and J. A. Copeland, “Reinforcement learning for repeated power control game in cognitive radio networks”, Selected Areas in Communications, IEEE Journal on, vol. 30, no. 1, **(2012)** January, pp. 54-69.

[9] M. Raiss El Fenni, R. El-Azouzi, M. El-kamili and K. Ibrahimi, “Dynamic spectrum allocation based on cognitive radio for qos support”, Proc of The 13-th ACM International Conference on Modeling, Analysis and Sim- ulation of Wireless and Mobile Systems, Bodrum, Turkey, **(2010)** October 17-21.

[10] X. Chao, M. Miyazawa, M. Pinedo and B. Atkinson, “Queuing networks: customers, signals and product form solutions”, The Journal of the Operational Research Society, vol. 52, no. 5, **(2001)** May, pp. 600-601.

[11] K. S. Narendra and M. A. L. Thathachar, “Learning Automata: An Introduction”, Engle- wood Cliffs: Prentice Hall, **(1989)**.

[12] D. Fudenberg and J.Tidole, “Game Theory”, the MIT Press, Cambridge, **(1992)**.

[13] K. S. Narendra and M. A. L. Thathachar, “Learning Automata: An Introduction”, Englewood Cliffs, NJ: Prentice-Hall, **(1989)**.

[14] P. S. Sastry Mandayam and A. L. Thathachar, “Networks of Learning Automata: Techniques for Online Stochastic Optimization”, Business & Economics, 268 pages, **(2004)**.

Authors

Mohammed Raiss El Fenni, he received the Ph.D degree in computer sciences from university of Avignon (France) and the Mohamed V university (Rabat,Morocco) in 2012. Currently, he is a professor at HECI business school in Morocco. His research interests include performance evaluation, stochastic learning, networking games and optimal control.

Mohamed El kamili, he received the Diplome d'études Supérieures Approfondies (2000) from the Mohammed V University (Rabat, Morocco). In 2005, he received the Ph.D degree in Networking and Computer Sciences from Mohammed V University (Rabat, Morocco). He acts as a reviewer for professional publications, prestigious international journals and international conferences, as ICC, Globcom. He has/is been involved in several national and international/European projects. Currently, he is a full-time associate professor at the Sidi Mohammed ben Abdellah University Fez Morocco. His current research interests include performance evaluation, protocols design, ad hoc networking, DTN and WSN, flexible radio, stochastic learning and networking games.