QUID 2017, pp. 2609-2619, Special Issue N°1- ISSN: 1692-343X, Medellín-Colombia

PRESENTING A LEARNING AUTOMATA-BASED SPECTRUM-SHARING METHOD IN A COGNITIVE RADIO NETWORK

(Recibido 05-06-2017. Aprobado el 07-09-2017)

Reza Sheikhi
Master's student of IT, Ghazvin Branch, Islamic Azad University, Ghazvin, Iran
Email: rzsheikhi247@gmail.com

Ali Naseri
Associate Professor, Imam Hossein University, Tehran, Iran
Email: iustnaseri2011@gmail.com

Abstract.The main objective of this study is to use the learning automata and Time Division Multiplex to reduce interference and improve the throughput in cognitive radio networks. In this method, in addition to the throughput, the cost of operation has been considered for secondary users. The proposed method is a new channel assignment process, which updates the probability of free channels at subsequent intervals, according to the environmental response using the learning automata, and increases throughput by decreasing the interference. For this purpose, three linear models on learning automata including L_{R-P}, L_{R-I} and L_{ReP}, were modeled using MATLAB software for access to the spectrum. The results of this simulation in the flexible environment of multiresponse learning automata showed that L_{R-I} model is an optimal method compared to other models. In the event that the primary user traffic model changes, L_{R-I} model cannot adapt itself to the new environment due to the lack of a penalty parameter. This will make it remain in the previous selection and fail to adopt its probability function according to the primary user traffic model. These problems make it possible to use the L_{ReP} model in a dynamic environment. This model pays neglible attention to penalty from the learning automata. In fact, L_{ReP} model acts like L_{R-I}, with the difference that it can adapt itself to the new environment. L_{R-P} model has severe fluctuations in updating its probability function, due to the equivalence of the reward and penalty rate. This makes it possible for each time slot to give a greater chance of choosing different channels and throughput decreases by increasing the number of switching channels.

Keywords: Cognitive radio, Secondary Users, Spectrum-Sharing, Multiresponse Learning Automata, Rules method

Citar, estilo APA: Sheikhi, R. & Naseri, A. (2017). Presenting a learning automata-based spectrum-sharing method in a cognitive radio network *Revista QUID (Special Issue),* 2609-2619.

1. INTRODUCTION

The radio spectrum is a very important source of wireless communications; however, this spectrum is not fully utilized, and the parts are also widely exploited by users. In the absence of spectral resources, we should seek intelligent radios called cognitive radios, and therefore, the embedded intelligence can instantaneously adapt its own conditions and parameters, such as transmit power, modulation order and type, coding type and rate, carrier frequency, bandwidth and the number of packages and frames with surroundings [A. He, S. Member, K. K. Bae, T. R. Newman, J. Gaeddert, K. Kim, R. Menon, L. Morales-tirado, J. J. Neel, Y. Zhao, J. H. Reed, W. H. Tranter, and L. Fellow . 2010]. Therefore, the need for spectrum sharing is felt perfectly. In this sharing scenario, this telecommunication band can be assigned to users under the secondary user category, when authorized users are not available, or it can be used, when the power of these signals in the receiver of the primary or authorized users is lower than the specified level, with the expansion of the secondary transmitted signals in a relatively broad bandwidth, and does not affect the throughput of primary user data (referred to underlay scenario [J. Oh, W. Choi . 2011]. The interference of these secondary users with the primary users is one of the main debates.

Since, all secondary users have the right for equal use of the spectrum in the unlicensed band, the main task of a cognitive radio network is to coordinate communication between them in an efficient and fair manner.

In the licensed spectrum, the primary network units have the exclusive right to communicate in specific frequency bands, while secondary units in the cognitive radio network only can occupy the licensed spectrum when the spectrum is not used by primary users. Therefore, the main task of a cognitive radio network is to identify the presence of primary users correctly. In addition, secondary units should also have the ability of continuous and uniform communication, even when forced to leave the channel in the presence of a primary user.

A well-designed cognitive radio network should pay attention to the quality of service of secondary users, in addition to addressing the degree of the interference in the primary units. In order to accomplish this, it is necessary to determine the spectrum management methods in a new way. Therefore, the necessity and need mentioned above is expected that the proposed systems will be a new step in solving the problems.

In this research, a learning-based approach is presented in which one or more secondary users can learn the primary user traffic model and use the most of the existing channels; while the secondary users should not interfere in the work of primary users. In the proposed method in this study, the objective of increasing the throughput is provided with the help of learning automata to identify the secondary users of the environment and considering the method of time division multiplex (TDM).

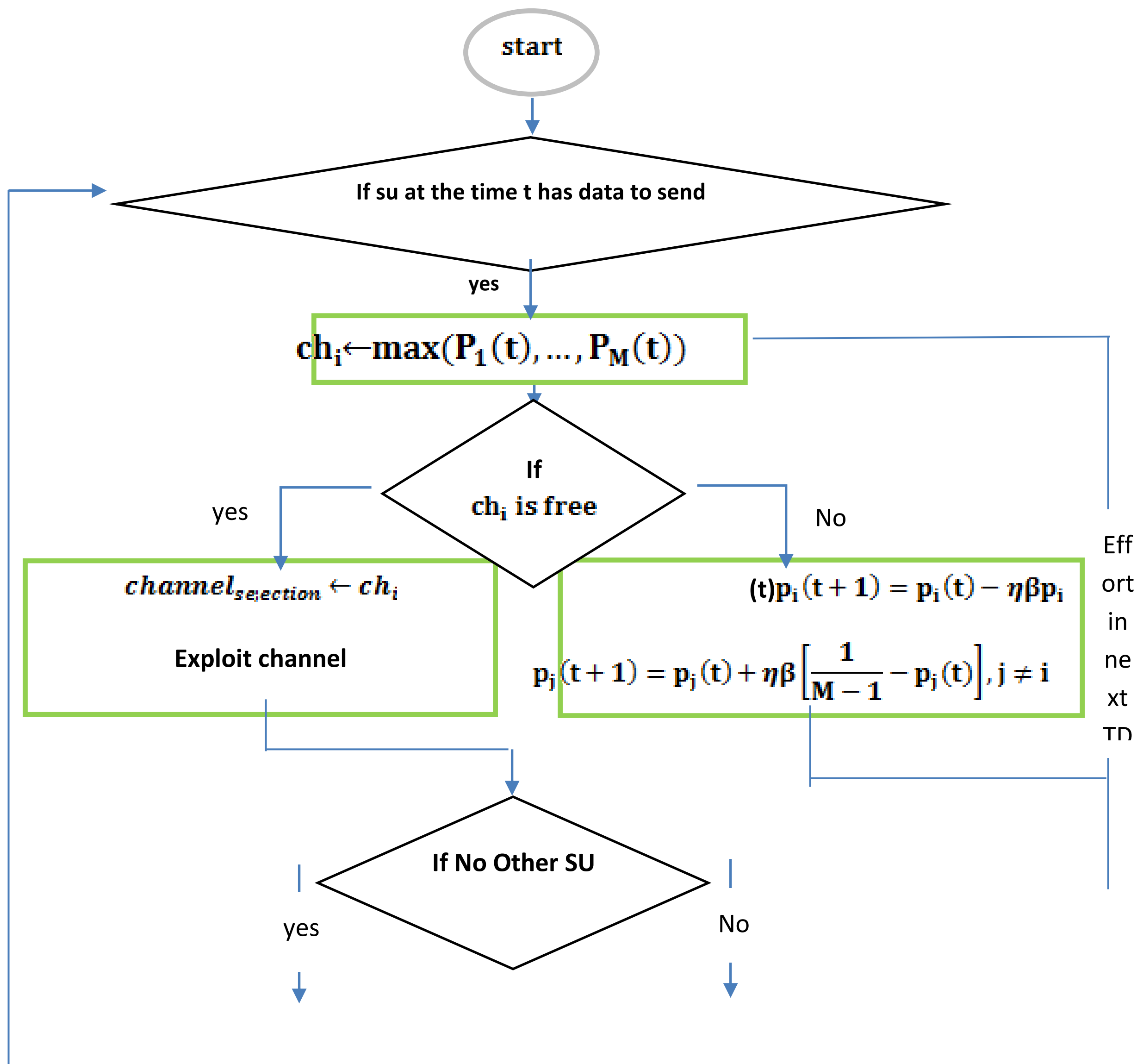
In fact, the presented method is a channel allocation process, which updates the free channels probability at subsequent intervals, given the secondary user environmental response, equipped with the learning automata. More precisely, in contrast to the previous methods, the new method learns the behavior of the environment and the use of the primary user for each node that wants a reliable communication in adaptive way, and records it in its system memory. Also, the proposed method provides a real and near-operational model for sending and receiving environment.

2. THE PROPOSED METHOD USING MULTIRESPONSE LEARNING AUTOMATA (MRLA)

Learning automata is a machine with limited states, which can perform a limited number of actions. Each selected operation is evaluated in a randomized environment and gives a response to the learning process. The learning automata use this response and choose its action for the next step. During this process, the learning automata learn how to choose the best practice among the authorized actions. The methods used by learning automata are slow learning, and resetting parameters requires time to learn [Kumpati S. 1974].

The proposed method is based on reference [Bizhani, H., & Ghasemi, A. 2013], according to which secondary users are trying to access channels under the primary user license. The purpose of using MRLA in this method is to reduce the collision between primary and secondary users, as well as among secondary users. Because with the variety of responses received from the environment (α_1 and α_2 and β), the secondary user has high flexibility in choosing the correct decision. Note that interference between the primary and secondary users is more important than the interference between the secondary users and, for this purpose, a larger reward is considered for the non-interference mode between the primary and secondary users, and also, the important advantage of this method to estimate filling or free channel is very effective in environments in which the user does not have accurate channel information and can only collect information from the environment by performing a test and error operation, that is consistent with the algorithm's learning property.

This learning model ultimately makes accurate recognition of channels and increases the chance of successful access to channels. The magnitude of the probabilities and their fluctuations in the long periods of time depends on the choice of α and β values, which take different values in different linear models. The proposed flowchart is as follows:



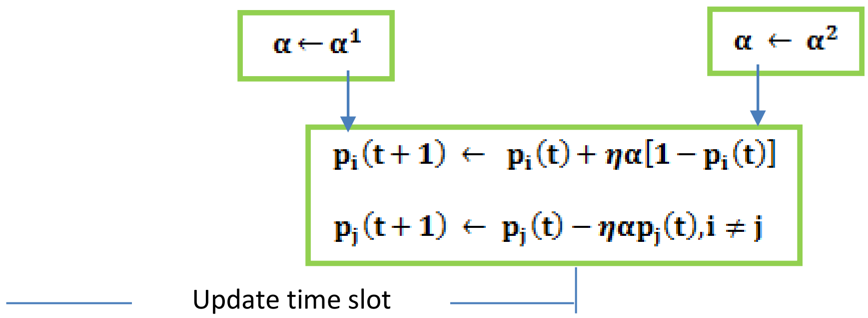


Chart 1 - Flowchart of the Proposed Method

2.1. Proposed Method System Model

In this system, the random variable of $\zeta_i(t)$ is defined. If in the time slot of t, channel i is free, the value of this random variable is equal to one, otherwise, when the channel i is occupied, its value is zero. Note that the secondary users are unaware of the probability vector value of primary user channel availability, ie, $\theta = (\theta_1, \ldots, \theta_M)$. In a multi-user model, we should consider the interference between secondary users in channel selection. For this purpose, the variable, $C_i(t)$, is defined as the number of secondary users that select the channel t in t time slot. When there is no interference in the choice of channel between the secondary users, $|C_i(t)| = 1$ and the selected channel i is free, $|\zeta_i(t)| = 1$, the secondary user can send the entire bandwidth over the selected interval, and if the selected channel i is occupied $|\zeta_i(t)| =$, the user will wait until the next time slot, according to TDM. In the next slot, he again tests his chances for channel selection. On the other hand, if the channel is free, but there will be interference with other secondary users $|C_i(t)| > 1$, there will be no successful sending on channel i and the bandwidth in that time slot does not belong to any of the secondary users. $NonCol_i(t)$ Variable is a Bernoulli variable, and when there is no interference in the choice of channel i between the secondary users in the t time slot, its value is equal to one and otherwise is equal to zero. Therefore, if we consider the channel bandwidth, B, the maximum number of bits sent by the secondary user n during the T time slot is calculated with (1).

$$w_n = \sum_{t=1}^{T} B \,.\, \zeta_i(t) NonCol_i(t) \quad (1)$$

In fact, the main purpose is to provide a way to maximize the total throughput mathematical expectation for all users. This equation is shown in (2).

$$w_N = \sum_{n=1}^{N} w_n = \sum_{n=1}^{N} \sum_{t=1}^{T} B \zeta_i(t) NonCol_i(t)) \quad (2)$$

If the user n selects i channel, which is full, and then selects another channel, the operational overhead will be increased and subsequently, the throughput will be reduced. Therefore, the cost of channel switching should be considered due to mistaken selection. To calculate this cost in the proposed method, the sw variable is used, which increases for each switch in the channel. According to this, $sw_n(t)$ is the total cost of the channel switching, which is calculated by the secondary user n during the t time slot, as follows:

$$sw_n(t) = c \sum_{i=1}^{M} S_{n,i}(t) \quad (3)$$

So that c is the channel switching cost, and $S_{n,i}(t)$ is the number of times that the secondary user n switches from each channel to channel i during the t time slot and M is the total number of channels available. Pseudocode below shows the calculation of the channel switching number in the algorithm process, for channel i.

M(0) =0 , $s_i = 0$

For t=1 to T do {

i=The selected channel based on P(t)

M(t)=i

If M(t)≠M(t-1)

$=s_i + 1s_i$

If channel i is free: Exploit &reward

elsepenalty}

As is seen in the above pseudocode, for the t-run time of the algorithm, the selected channel is stored according to the probability vector $P_{(t)}$ inside $M_{(t)}$, which represents the selected channel at time t, and it is compared each time with the selected channel of the previous step (t-1) and if there is an inconsistency, it is understood that the selected channel of the user has changed compared to the previous time, then one unit should be added to the switching cost (s).

Note that the rate of channel switching is very important in learning methods, because of the use of reward and penalty at each stage and the fluctuation of the probability function at each stage, different channels can be selected, and according to what was mentioned, in addition to increasing the overhead cost resulting from the implementation of the algorithm, it would also reduce the throughput.

2.2. Channel Selection Steps in The Proposed Method

{ Presenting the primary values (α) and (β), based on standard linear models $L_{R-I}, L_{R-P}, L_{Rep}$

Considering the probability of $\frac{1}{M}$ for each channel, initially, for the T-Run time of the algorithm:

The secondary user n, which has data for sending at t time, performs the following steps to access the channel and update its probability functions:

-In TDM time slots, the channel i is selected based on the second user probability function, which is initially equal to $\frac{1}{M}$.

-Full or free channel i is checked

Based on the MRLA method, "the response which is received from checking channel:"

1. If the selected channel i, is free:

Uses the channel and checks the following condition for updating its probability function:

If not occupied by another secondary user: r = 1

If occupied by another secondary user: r = 2

And then updates its probability function for channel i and other channels (j) [5]:

$$P_i^n(t+1) = P_i^n(t) + \eta\alpha^r[1 - P_i^n(t)]$$

$$P_j^n(t+1) = P_j^n(t) - \eta\alpha^r P_j^n(t), j \neq i$$

2. If the selected channel i is not free:

This means that the choice is wrong and must have a penalty [5]:

$$P_i^n(t+1) = P_i^n(t) - \eta\beta P_i^n(t)$$

$$P_j^n(t+1) = P_j^n(t) + \eta\beta\left[\frac{1}{M-1} - P_j^n(t)\right], j \neq i$$

Therefore, the media are occupied and needs to retry to receive a TDM slot}

2.2.1. Describe The Steps to Select the Channel in The Proposed Method

As seen in the implementation of the proposed method, the reward and penalty parameters are initialized based on linear models and their expressed values in the first line. Since, at first, the learning automata have no channel information, their probability functions is $\frac{1}{M}$, and M is the same number of available channels. Then, at each interval, the secondary user chooses i channel based on its probability function. If it is free, he uses the channel and checks that it does not collide with any other secondary user. If there is no collision with a secondary user, he sets the value of parameter r equal to one, and if there is a collision, he sets the value of its parameter r equal to two, and eventually, the reward function will be updated for the secondary user accordingly. Note that when the value of r is equal to 1, the reward function in accordance with the above algorithm will have a larger value (α^r), in this case, the value of the reward function for the n user increases in the selected channel, due to the lack of collision with the primary users so that the channel has a higher chance of choosing in the next time interval. If the channel is occupied, the intended user must be penalized and its probability function will be updated according to (3), so it gives a lower selection chance for the intended channel in the next time interval.

2.3. Analysis of The Proposed Method

The proofs of the discussion and relations expressed in this section are given in references [Narendra, K. S., & Thathachar, M. A. L.1989]. An important issue that needs to be addressed about the use of learning automata to identify the channel is the convergence of the functions used by each secondary user. Because, in the reinforcement learning model, the functions of each secondary user vary each time with a reward and penalty, it is important that the rewards and penalties are controlled and expressed in the form of a standard model. Because if the rewards and penalties are determined randomly and without precision, after the passage of time, the value of these probabilities passes through the zero and one intervals and causes problem for the selection of the channel, by losing the concept of the access probability. For this purpose, the reward and penalty amounts used in the proposed method are expressed in the form of standard linear models L_{R-P}, $L_{R-I}$and L_{Rep}, in order to solve the convergence problem of the proposed method.

Therefore, the convergent nature is directly related to the type of learning model used in the learning automata. For example, the linear models L_{R-P} and L_{Rep}, do not have an absorbing state[1] in probability functions due to the effect of the two rewards and penalty, and converge to a constant distribution.

In the L_{Rep} scheme used in the model, the algorithm converges to a normal distribution with the value (4), which is independent of the primary value of P (0):

$$P_i = \varepsilon \frac{(1-\theta_H)}{(M-1)(\theta_i+\theta_H)}, i = 1,2,\ldots,M \text{ and } i \neq H \tag{4}$$

$$P_H = 1 - \sum_{i \neq H} P_i$$

[1] In the theory of probability, the Markov chain, which in each case can reach an absorbing state, is called the absorbing Markov chain. The absorbing state is a condition that once you enter it, you cannot get out of it.

So, θ_i is the probability that the channel i is empty and H is the channel index with the most likely to be empty. We can show a chance to get rewards and penalties, which are equal to the free and the occupied probability of the selected channel, with the following equations:

$$d_i = \Pr[Res(t) \epsilon \{\alpha^1, \alpha^2\} | a(t) = a_i] = \theta_i \tag{5}$$

$$c_i = \Pr[Res(t) = \beta | a(t) = a_i] = 1 - \theta_i \tag{6}$$

Res (t) is the response of the environment to the selected action i in the time interval t, the sum of the reward and penalty probabilities is equal to the constant value (7) and converges to a random variable with the value of (8):

$\forall i = 1,2,\ldots,M$

$$\alpha_i^1 d_i^1 + \alpha_i^2 d_i^2 + \beta_i c_i = const \tag{7}$$

$$\lim_{t\to\infty} E[P_i(t)] = \frac{1/\beta(1-\theta_i)}{\sum_{j=1}^{M} 1/\beta(1-\theta_j)} = \frac{1/(1-\theta_i)}{\sum_{j=1}^{M} 1/(1-\theta_j)} \tag{8}$$

Equation (8) is for the model $MRLA_{R-P}$ because in this linear model, reward-penalty is $\alpha_i^r = \beta_i^p$. While L_{R-I} converges to an optimal action, due to the lack of a penalty factor, with probability one. Convergence is absorbing states in fully-exponential models, such as L_{R-I}, and their probability vector converges to one over time. Contrary to the two previous models, this model is dependent on the initial value of $P(0)$ and, due to the absorption property, cannot find the appropriate new channel and adapt it to the dynamic conditions of these types of channels, therefore it is not suitable for dynamic cognitive radio networks.

2.4. Evaluation Criteria for The Proposed Method

The accurate estimation of network performance is a critical issue. For this reason, the selection and design of performance parameters and evaluation functions is an important issue in cognitive radio networks. Because of the dynamic nature of these networks, intelligence network designers are trying to get a better understanding of the internal relationships between targets, performance parameters, evaluation functions, network performance, and links and operational environments. Network performance management is in order to optimize the methods so that the network can increase capacity and reduce delays, regardless of the bandwidth availability and the failure of the sending.

One of the most important performance evaluation criteria, which are presented in most of the methods as the main goal for improving algorithms, is throughput, which seeks to improve it for secondary users who use the channel. Various issues, such as the multiplicity of secondary users and the computational cost of the proposed algorithms are as an obstacle to increasing throughput. In this research, the proposed method of throughput is used to evaluate the efficiency. Various factors are involved in the amount of throughput, of which the most important are the amount of channel switching during the execution of the algorithm for each wrong channel selection and the number of secondary users who are in the network and intend to use the channel. These two factors are investigated in this method and ultimately, we calculate the throughput for the proposed algorithm.

Equation (9) is used for calculating the channel switching cost and equation (10) is used for calculating the throughput in simulation.

$$sw_n(t) = c\sum_{i=1}^{M} S_{n,i}(t) \tag{9}$$

So, c is the channel switching cost, and $S_{n,i}(t)$ is the number of times a secondary user n switches during t time slot from each channel to channel i and sw (T) is the variable that calculates the total number of switching channels.

$$W = \sum_{n=1}^{N} w_n = \sum_{n=1}^{N}\sum_{t=1}^{T} B\zeta_i(t) NonCol_i(t) \tag{10}$$

B is the channel bandwidth used, $\zeta_i(t)$ is the variable that takes one, in the absence of a channel in the t slot, $NonCol_i(t)$ is the Bernoulli variable and if there is no interference between the secondary users in interval t, it takes 1, i is the selected channel index, n is the secondary user index and W is the total throughput .

2.5 Simulation Parameters

The parameters and values used in the simulation are presented in Table 1 below:

Table 1: Parameters Used in Simulation

Parameters	Value	Definition
M	10	Available channels that can be accessed by secondary users
N	20	Number of secondary users
T	4000 timeslot	Number of time intervals
C	1	Channel switching cost
r	2	Effective amount to calculate the reward factor, in case of interference with the other secondary user
P	0.1	Initial probability, for access to any channel
D	Random	Distribution of user input
T	5 users	The threshold value for the presence of the secondary users in the time slot

2.6. Simulation Space with Considering the Assumptions

A cognitive radio network with 10 channels has been considered in two single-user modes for calculating the number of switching channels and the behavior of the proposed algorithm for different and multi-user reward and penalty amounts, in order to calculate the behavior of the algorithm in the multiplicity of secondary users and its effect on throughput. The secondary user is trying to use a free channel without interfering with the use of information collected from the environment. It is assumed that the number of secondary users who need to use the channel at any time interval is less than the number of channels available (N ≤ M). Assuming a slotted system, users in the network operate with the same time slot model and at the beginning of each time slot, $t \in \{1,2,..,T\}$, the secondary user that has data to send, selects a channel for sensation and, based on the result of the sensation, the channel is available for access. Assuming that users divide their data into frames corresponding to the time slot on the channel before sending, as well as the secondary users can correctly feel the channels, by means of the mechanism, and there is no error in sensation. In other words, the secondary user n can correctly identify the presence or absence of the primary user on the channel i. In general, in each time slot, the channel i is free with probability θ_i and occupied with probability $1-\theta_i$. Note that secondary users are not aware of the actual value of the probability vector of the primary user channel availability, ie, $\theta_1, \ldots, \theta_M$. The simulation is done for two static and dynamic environments, in order to examine the behavior of the time algorithm that the input distribution of the primary users is changed. In the first step, the input distribution of primary users to the system in all time slots is considered to be fixed, and in the next step, the input distribution of primary users to the system in the time slot of 2000 is changed, so that the behavior of the proposed method in various linear models is measured. In order to do simulation, the MATLAB software is used.

3. REVIEW THE RULES METHOD AS A BASIS FOR COMPARISON WITH THE PROPOSED METHOD

The Rules [H. El Gamal and H. V. Poor . Feb.2011], which is a subset of learning-enhancing methods, is a solution to these problems, in order to ensure that, in equal conditions, the method of learning from the environment and the enhancement of information chooses the appropriate action at each step. The objective is increasing the throughput and it is considered as the basis of comparison with the proposed method.

In this way, the secondary user will discover the probability of channel availability. Initially, the conditions under which channels are likely to be known by secondary users are investigated. To avoid the selfish behavior of secondary users, a model is introduced using game theory. Given the fact that secondary users are not aware of the availability of channels, an optimal algorithm has been introduced. The proposed algorithm was introduced in a multi-user model to access a channel (Rule3), in which the secondary user has two vectors X and Y, and X_i is the number of time slots, with which channel i is selected and empty. Y_i, is the number of times that channel i has been selected. This method works as follows:

1. Initialization: First, all channels are sensed and the values of the vectors X_i and Y_i are determined.

2. After the initialization step, at the beginning of the time slot j, the secondary user calculates an estimation of $\hat{\theta}$, which is the probability of the free channel, using $\hat{\theta}_i(j) = X_i(j)/Y_i(j)$ and chooses the channel based on it.

3. Then the X and Y vectors are updated based on the new values received from the environment. The difference between the proposed algorithm and the proposed method of the present study is the degree of flexibility in exchange with the environment and the amount of penalty and reward factor. The use of multiple responses from the environment, rather than receiving a response in the proposed method has made it more dynamic in interacting with the environment and experiences closer behavior to the operating environment, as well as in the reference method [H. El Gamal and H. V. Poor . Feb.2011], changing the selected channel will be done at a low speed, because of the lack of a deterrent factor in choosing the channel, in the event of a wrong choice, which will result in the throughput being dropped.

In the proposed method, using the learning automata, the attempt is made to increase the throughput, which is an important criterion of performance measurement, and the proposed method has a better throughput in the same operating conditions relative to [H. El Gamal and H. V. Poor . Feb.2011].

4. SIMULATION RESULTS

4.1. Reviewing The Cost of Channel Switching

In Fig. 2, each wrong channel detection cost and the channel switching of the secondary user is calculated in different linear models and it is compared with the proposed method in [H. El Gamal and H. V. Poor . Feb.2011].

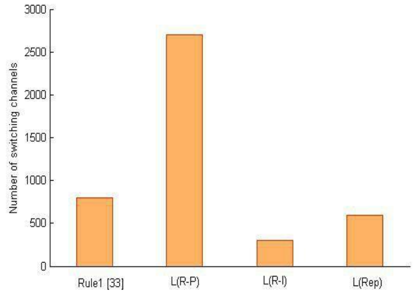


Chart 2 - Channel Switching Number Between Proposed Method in Various Linear Models and Rule1 [H. El Gamal and H. V. Poor . Feb.2011]

As seen in Chart 2, in the L_{R_I}method, we will have less channel switching because the penalty factor is zero. Because the secondary user, at any time period that chooses the channel, is not penalized if the channel is occupied, therefore, it switches to the other channel with a very small probability, and this causes the number of channel switching in this model has the lowest amount.

The probability vector in this model, in cases where the channel i is not free, is given as (4-5) due to the value of β = 0:

$$P_i^n(t+1) = P_i^n(t) - \beta P_i^n(t) \rightarrow P_i^n(t+1) = P_i^n(t) \tag{11}$$

As can be seen, the probability vector value for the n-th secondary user does not decrease in the next time slot and is equal to the previous value. In the L_{Rep} and L_{R-p} models, the existence of a penalty factor (β) causes more fluctuations in the values of probability functions, and these changes make the secondary user tries more channels to choose, which increases the number of channel switching. This number of switching in the L_{R-p}model is high due to the high penalty factor and it is low in the L_{Rep} model, due to the small penalty factor ($\beta \ll \alpha$). The switching cost in the L_{ReP} model is greater than the $L_{R\text{-}I}$, because, unlike the $L_{R\text{-}I}$, the L_{ReP} model gives a chance of selection to other channels as well. $L_{R\text{-}P}$ has a different behavior compared to other linear methods. Since α = β in this model, the probability updates the choosing actions for the reward and the penalty response at the same rate. This will reduce throughput, because for every attempt failed in accessing to the channel, it switches the selected channel more likely, which leads to a greater number of switching channels, and the throughput is reduced.

4.2. Examining The Throughput in The Static Environment

L_{R-I} is a very expedient model [5]. The secondary user who uses this model to access the channel can find the optimal channel after some time slots. In other words, by learning the traffic model, the primary user can detect the best channel from the other channels.

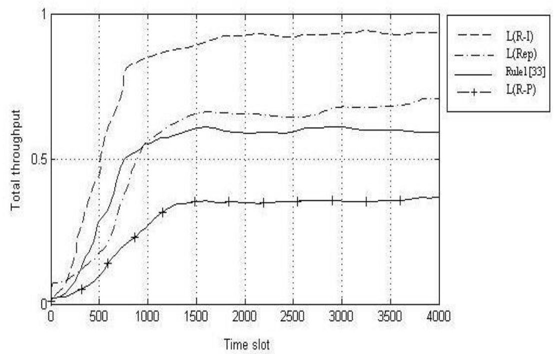


Chart 3: Comparison Of Total Throughput, Between The Proposed Method And Rule1 [H. El Gamal and H. V. Poor . Feb.2011] In The Static Environment

According to Chart 3, linear models $\mathbf{L_{Rep}}$ and $\mathbf{L_{R-P}}$ have lower throughput than $\mathbf{L_{R-I}}$ model due to more channel switching. This method is very suitable for a situation where primary user traffic is constant, because the channel finds high with the probability of being free and selects it to the end. $\mathbf{L_{R-P}}$ is a model in which the penalty factor is very high, and for this reason, its probability function has high fluctuation compared to other methods, and this causes the throughput to be reduced?

4.3.3. Examination of Throughput in A Dynamic Environment

To examine the proposed method in linear models, we change the traffic of the primary users, which is the probability that the channels are free in the 2000-time interval, to examine the behavior of the proposed method in real environments, which channel traffic is constantly switching. As illustrated in Chart 4 and the analysis of the linear model $\mathbf{L_{R-I}}$, the problem of the $\mathbf{L_{R-I}}$ method is that if the network is dynamic and the traffic model of the primary users' changes over time, it causes problems, because in this case, with the change in the possibility of access to channels, the secondary user using this model to identify the channel, cannot find the new channel, and the throughput is severely reduced. To solve this problem, we can use the $\mathbf{L_{Rep}}$ model. This method acts like $\mathbf{L_{R-I}}$; except that in the L_{ReP} model, the penalty factor is also very small compared to the reward. The L_{ReP} reinforcement model is $\epsilon - optimal$ [5]. It can be seen that L_{ReP} works well in a dynamic environment, and can quickly adapt to a dynamic traffic environment.

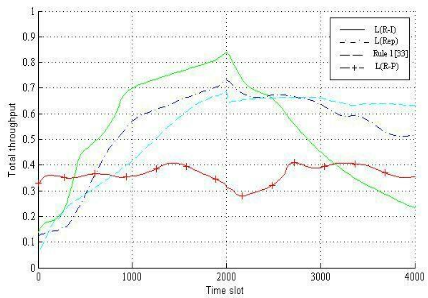


Chart 4: Comparison of Total Throughput in The Proposed Method and Rule1 [H. El Gamal and H. V. Poor . Feb.2011] in A Dynamic Environment

4.4. Examining The Effect of Secondary Users Rate on the Throughput

As suggested in the proposed method, it is assumed that, according to the input distribution of secondary users to the system, the number of secondary users who intend to use the system during the time interval t is less than the number of channels ($N \leq M$). Otherwise, there should be a mechanism to avoid additional users input and managed. Chart 5 shows the number of secondary users increases in the system at each stage.

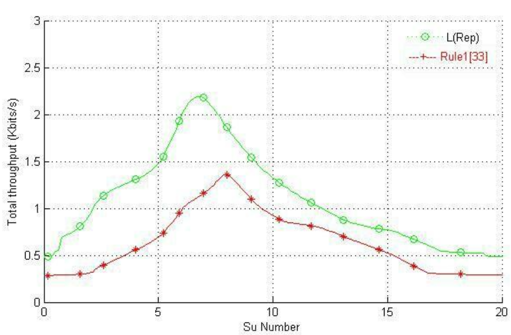


Chart 5: Comparison of The Proposed Method with Rule3 [H. El Gamal and H. V. Poor . Feb.2011] With Increasing Number of Secondary Users

As you can see, as long as the number of secondary users using the channel is less than 10, this method can be an effective way to select the channel, while, with the increase in the number of secondary users in the system for access to the channel, the throughput is reduced, because the number of successful channel selection for each secondary user is reduced for the channel selections relative to the total requests. Of course, this is due to the large number for secondary users, since the number of available channels is limited and the throughput rate for secondary users who intend to use the channel decreases, due to the channel constraints, channel filling, and collision between other users.

5. CONCLUSION

The objective of this study was to study learning automata method to access the spectrum. In this regard, a new method of access control was introduced to increase the network's throughput and to discuss the total throughput of the secondary users and the amount of channel switching in the network.

In this research, learning automata was used in a multi-response environment to obtain channel status that each secondary user's functions are updated to access the channel at subsequent intervals based on the environmental response. The proposed method was expressed using the algorithm in [7], and it was discussed about the proposed method in terms of convergence in three linear models of $\mathbf{L}_{Rep}$, $\mathbf{L}_{R-P}$ and $\mathbf{L}_{R-I}$, among which methods $\mathbf{L}_{Rep}$ and $\mathbf{L}_{R-P}$ can adapt themselves to the dynamic environment, but the $\mathbf{L}_{R-I}$ model chooses the optimal channel because of its absorption property, and remains on the same channel, because in this model, the algorithm is not penalized by a collision with the primary users due to the absence of a penalty factor, and does not change its own choice in the next time interval based on the wrong choice, which is why it is not suitable for cognitive radio networks. Then, it uses the method in [H. El Gamal and H. V. Poor . Feb.2011], which is a reinforcing model, in which several secondary users are trying to access a channel, which is the basis for comparing the proposed algorithm of the present study.

In the following, three learning automata linear models, called L_{R-P}, L_{R-I} and L_{ReP}, were used to access the spectrum. The results of this simulation in the flexible environment of the multi-response learning Automata, showed that L_{R-I} model is an optimal method compared to other models. In the event that the primary user traffic model is changed, L_{R-I} model cannot adapt itself to the new environment due to the lack of a penalty parameter. This results in staying in the previous choice and fail to adopt its probability function based on the primary user traffic model.

These problems cause to apply L_{ReP} model in a dynamic environment. This model of learning automata also concerns a very small effect of penalty. In fact, the L_{ReP} model acts like the L_{R-I}, with the difference that it can adapt itself to the new environment. L_{R-P} model has severe fluctuation in updating its probability function, due to the equivalence of the reward and penalty rate. This makes more chance to select different data channels in each time slot and with increasing the number of channel switching, the throughput reduces.

6. RESEARCH SUGGESTIONS

Following, we present some suggestions:

- Adding channel access section by secondary users, in the cluster template according to the model presented in reference [Bolívar, N., Marzo, J. L., & Rodríguez-Colina, E. 2010].

- The use of other criteria, such as secondary users' signal strength, hand off, users distance from each other and the amount of delay in the sending, as rewards and penalty responses to the environment by the MRLA.

- Examining different rates of rewards and penalties in the MRLA environment and its effects on the environment.

- The use of multiple learning automata for each secondary user, which optimizes various parameters of the environment in parallel.

- Analyze and use the cellular learning automata in different conditions to access the channel.

- Study the behavior of the non-linear models of learning automata with access to dynamic cognitive radio spectrum.

- Use of justice index in a multi-user environment to observe the extent of each user's use of the channel.

- Examining admission control of secondary users to block some secondary users, when the number of secondary users is greater than the selected channel, which prevents interference from secondary users.

REFERENCES

A. He, S. Member, K. K. Bae, T. R. Newman, J. Gaeddert, K. Kim, R. Menon, L. Morales-tirado, J. J. Neel, Y. Zhao, J. H. Reed, W. H. Tranter, and L. Fellow . “A Survey of Artificial Intelligence for Cognitive Radios,” . IEEE vol 59, no. 4, pp. 1578–1592.2010

J. Oh, W. Choi . "A hybrid cognitive radio system: a combination of underlay and overlay approaches".*Proc. IEEE Int. Conf. Veh. Technol.*, pp. 1-5. 2011

Bolívar, N., Marzo, J. L., & Rodríguez-Colina, E. “Distributed Control Using Cognitive Pilot Channels in a Centralized Cognitive Radio Network“.SixtAdvancedInternationalConferenceon Telecommunications, 3(3), 30–34.2010

Bizhani, H., & Ghasemi, A. “Joint Admission Control and Channel Selection Based on Multi Response Learning Automata (MRLA) in Cognitive Radio Networks“. *Wireless Personal Communications*, *71*(1), 629–649. 2013

H. El Gamal and H. V. Poor . Feb. “Cognitive Medium Access: Exploration, Exploitation, and Competition,” *IEEE Trans. Mob. Comput.*, vol. 10, no. 2, pp. 239–253. 2011

Kumpati S. Narendra, Senior Member .“Learning Automata A Survey,“ IEEE Transactions On Systems, Man, And Cybernetics, Vol. SMC-4, No. 4.July 1974

Narendra, K. S., & Thathachar, M. A. L.“Learning automata: An introduction“ . EnglewoodCliffs, NJ: Prentice Hall.1989