DOI: 10.21917/ijct.2013.0112

EFFICIENT SPECTRUM UTILIZATION IN COGNITIVE RADIO THROUGH REINFORCEMENT LEARNING

Dhananjay Kumar[1], Pavithra Hari[2], Panbhazhagi Selvaraj[3] and Sharavanti Baskaran[4]
Department of Information Technology, Anna University, MIT Campus, Chennai, India
E-mail: [1]dhananjay@annauniv.edu, [2]jayapavithra25@gmail.com, [3]panbhu2009@gmail.com, [4]Lolita.july@gmail.com

Abstract
Machine learning schemes can be employed in cognitive radio systems to intelligently locate the spectrum holes with some knowledge about the operating environment. In this paper, we formulate a variation of Actor Critic Learning algorithm known as Continuous Actor Critic Learning Automaton (CACLA) and compare this scheme with Actor Critic Learning scheme and existing Q–learning scheme. Simulation results show that our CACLA scheme has lesser execution time and achieves higher throughput compared to other two schemes.

Keywords:
Markov Decision Process, Reinforcement Learning, Q–learning, Actor–critic Learning, CACLA

1. INTRODUCTION

The radio spectrum is under a great demand due to the tremendous development in wireless technology. Regrettably the supply of spectrum has not met the requirement of emerging systems. The dearth in supply of spectrum is chiefly due to the ineffective, rigid and stable nature of the surviving spectrum utilization methods and emphatically not because of the scarceness of operable spectrum [1]. Hence, it is mandatory to improve mechanisms that provide effectual exploitation of these available spectrums. In order to enhance the efficient spectrum utilization, FCC promoted the opportunistic spectrum access (OSA) that allows secondary users to independently search for and exploit instantaneous spectrum availability. To realize OSA cognitive radio has been advocated by many. CRs are viewed as intelligent wireless communication systems that are capable of self-learning from their surrounding environment and auto adapting in their real time to improve spectrum efficiency with no interventions [2].

Research in protocol design [3, 4] has increased because of high interest in OSA where the MAC protocols provides the secondary users to explore and make use of the spectrum which is left unused in a way that constrains the level of interference to the primary users. A new analytical approach has been provided for the problem of spectrum agility [5] in which the over utilization of the spectral bands and underutilization are diagnosed and controlled by means of an analytical model. Spectrum sharing [6] plays a major role in communication which are analyzed and efficiently utilized by providing a statistical trade off. In [7] the opportunistic access is solved by using a slotted transmission protocol for secondary users by means of periodic channel sensing strategy. The spectrum sensing and accessing policies results in the interference among the primary users is overcome by assuming that these users have their distribution parameters as unknown random parameters[8].Liu and Zhao [9] reckoned a distributed secondary users environment where they communicate among each other without the transmission of information.

The activities of the Primary Users follow the Markovian process model. The OSA environment follows a distinct feature which makes it too hard to fabricate models which predicts the dynamics. So it is very important to develop techniques that can accomplish the same optimal behaviors without considering the models of the environmental dynamics. The reinforcement learning (RL) [10] is a sub–field of machine intelligence which is a basis of learning & interaction among the user and the environment. In [11], Pavithra Venkatraman introduced a type of scheme called Q–learning in cognitive radio to efficiently use the spectrum. Other learning schemes used in cognitive radio are discussed in [12]–[14]. A Nash equilibrium based Reinforcement Learning scheme for power control is designed with low implementation complexity for CR networks [12].The scheme does not require the interference channel and power strategy information among CR users. To solve the problem of distributed learning and channel access, a distributed learning with logarithmic regret is developed which achieves optimal cognitive system throughput [13]. The policies minimize the sum regret in distributed learning which is the loss is in secondary throughput due to learning and distributed access. Jayakrishnan Unnikrishnan [14] proposed RL–DSA for DSA in the on text of next–generation downlink OFDMA–based networks. RL–DSA maximizes a reward signal targeting an efficient spectrum usage with QoS assurance. Moreover, RL–DSA had shown best tradeoff between spectral efficiency, QoS fulfillment and fairness among different spectrum assignment strategies.

In this paper, we compare our proposed Reinforcement Learning scheme namely Continuous Actor Critic Learning Automaton (CACLA) scheme with Actor–Critic learning [10, 15] and existing Q–learning schemes in terms of execution time, bit rate and throughput. Simulation results show that our CACLA scheme achieves high throughput performance and less execution time compared to other two schemes.

The remainder of this paper is organized as follows. Section 2 describes the system architecture. Section 3 details the proposed RL schemes. Section 4 evaluates the performance of all the three RL schemes. Section 5 concludes this paper.

2. SYSTEM ARCHITECTURE

2.1 PROBLEM FORMULATION

We assume that the spectrum is divided into *m* bands and each band is associated with a PU [11]. Moreover, we assume that the SUs are non–cooperative. Therefore each SU maintains its own Q/V/P/A table. At each time step the SU reports to decision center which makes the decision regarding which band to switch over from the available spectrum opportunities (Fig.1).

Then the SU switches to that band and starts communicating on it. The identification of spectrum opportunities depends upon the quality metrics such as the SNR, band width, and probability of PU returning to band.

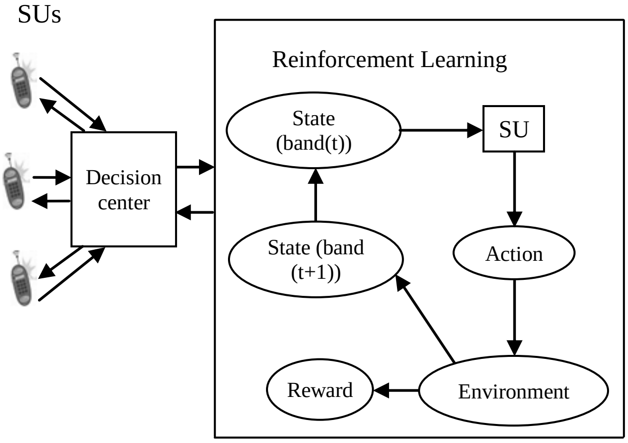

Fig.1. Reinforcement learning for SUs

The rest of this section describes about the Markov Decision process and the RL schemes.

2.2 MDP

Markov decision process (MDP), is defined as follows.

State Set S: S is made up of m states $\{s_1, s_2, \ldots s_m\}$. When the SU is in the state s_i then it means that the SU is in band b_i with minimum probability of PU returning to band b_i.

Action Set A: The number of actions is always equal to the number of bands. At each time step while in band b_i, the SU with the help of decision center explores by searching for new spectrum opportunities. The SU senses the bands $\{b_1, b_2, \ldots b_m\}$ sequentially, until it finds the first available band. After finding a band, the SU switches to and starts using it until the next time step.

Reward Function: The reward function r_{s_i,a_i} specifies the reward the SU obtains when taking action a_i from the action set A while in state s_i ε S. Actions are selected using ε – greedy exploration strategy which states that action which gives maximum Q/V value is selected with (1 – ε) + ε/m probability and any other action is selected with ε/m probability where m is the number of bands. The reward obtained by the SU depends on the actions chosen by other users. We assume each band b_i has its own bandwidth capacity v_i and when more than one SU uses the same band the bandwidth is equally divided among all the users who use the same band. For example if there are three SUs, namely A, B, C each taking an action i, j, k respectively, their reward of SU A, can be calculated as [16].

$$r_{ijk} = \begin{cases} \frac{v_i}{3} & \text{when } i = j = k \\ \frac{v_i}{2} & \text{when } i = j \neq k \text{ or } i = k \neq j \\ v_i & \text{when } i \neq j \neq k \end{cases} \quad (1)$$

The reward perceived by the SU also depends on the probability that the PU return back to each band and SNR.

2.3 Q–LEARNING SCHEME

At each time step, each secondary user uses the Q-learning scheme to select a spectrum band among all the bands. The aim of the SU is to learn a policy (band to be switched over) π: $S \rightarrow A$ for choosing the next action a_i based on its current state s_i that gives the maximum reward. A function Q: $S \times A \rightarrow R$ is defined for each state–action (s_i, a_i) pair as the maximum reward that can be achieved when taking action a_i from state s_i according to the ε–greedy exploration strategy. Hence, with the help of the Q–function, the SUs can optimally select actions that maximize Q_{s_i,a_i} at each state. The Q–learning algorithm works by selecting actions and observing the following state and the resulting reward. With this information, Q is updated via the following equation [16],

$$Q^{su}_{s_i,a_i}(t+1) = Q^{su}_{s_i,a_i}(t) + a[r_{s_i,a_i} - Q^{su}_{s_i,a_i}(t)] \quad (2)$$

where, 0 < α <1 is the learning rate of the secondary user group, and r_{s_i,a_i} is the reward function. Fig.2 provides a generalized flow of Q-learning algorithm.

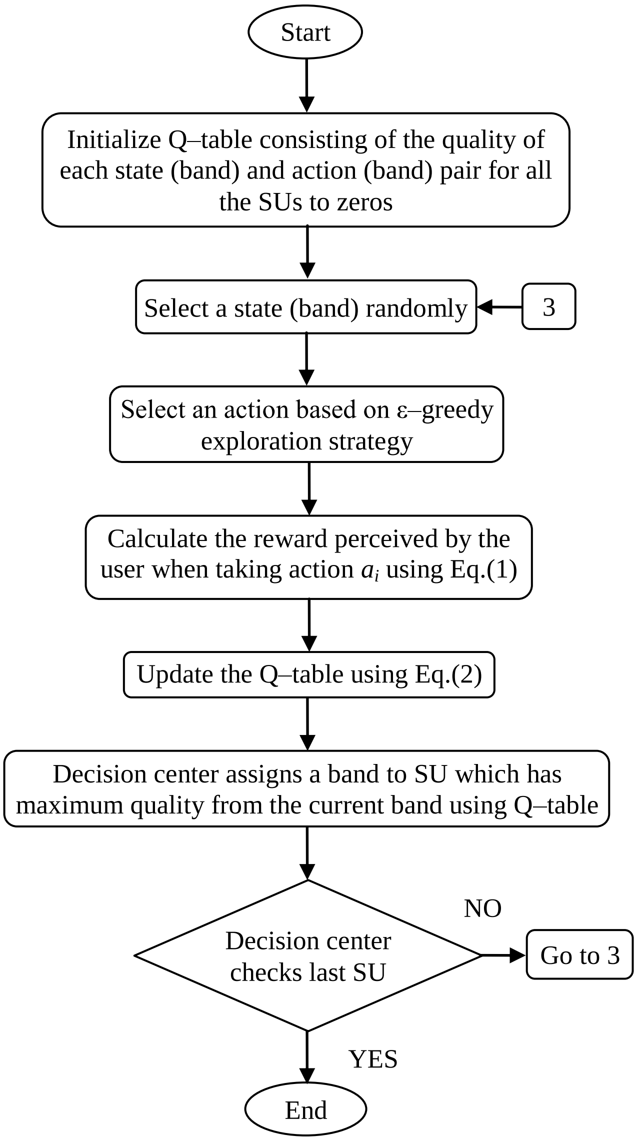
Start
Initialize Q–table consisting of the quality of each state (band) and action (band) pair for all the SUs to zeros
Select a state (band) randomly
3
Select an action based on ε–greedy exploration strategy
Calculate the reward perceived by the user when taking action a_i using Eq.(1)
Update the Q–table using Eq.(2)
Decision center assigns a band to SU which has maximum quality from the current band using Q–table
Decision center checks last SU
NO
Go to 3
YES
End

Fig.2. Q – Learning flow chart

3. PROPOSED SYSTEM

At each time step, each secondary user uses the Q-learning scheme to select a spectrum band among all the bands. The aim of the SU is to learn a policy (band to be switched over) π: $S \rightarrow A$ for choosing the next action a_i based on its current state s_i that gives the maximum reward.

3.1 ACTOR CRITIC LEARNING SCHEME

A function V: $S \rightarrow R$ is defined for each state–action (s_i, a_i) pair as the maximum reward that can be achieved when taking action a_i from state s_i according to the ε–greedy exploration strategy and a Preference table (P) which stores the preference of state dependent actions are defined for each SU. Actor critic learning scheme learns by selecting actions and observing the following state and the resulting reward. With this information, V is updated via the following equation,

$$v_{s_i}^{g}(t+1) = v_{s_i}^{g}(t) + a[r_{s_i,a_i} - v_{s_i}^{g}(t)] \tag{3}$$

where, $0 < \alpha < 1$ is the learning rate of the secondary user group g and r_{s_i,a_i} is the associated reward function.

Actor critic learning uses state values to update the preference table.

$$P_{s_i,a_i}^{u}(t+1) = v_{s_i}^{u}(t) + a[r_{s_i,a_i} - v_{s_i}^{u}(t)] + P_{s_i,a_i}^{u}(t) \tag{4}$$

Usage of state values speeds up the learning process. With the help of the P-table, the SUs can optimally select actions that have highest preference at each state. Fig.3 provides a flow chart of Actor-Critic Learning algorithm.

3.2 CONTINUOUS ACTOR CRITIC LEARNING AUTOMATON SCHEME

A function V: $S \rightarrow R$ is defined for each state – action (s_i, a_i) pair as the maximum reward that can be achieved when taking action a_i from state s_i according to the ε–greedy exploration strategy and a Preference table (P) which stores the preference of state dependent actions and an Action (A) table which stores the actions from each state (band) defined for each SU. Continuous actor critic learning automaton algorithm acquires knowledge by selecting actions and observing the following state and the resulting reward. With this information, V is updated via Eq.(5) as it is a variation of Actor Critic Learning scheme.

Continuous actor critic learning automaton uses state values to update the A-table. To update to these action values, one observes the update to the state value of the last state. If this update increased the value of the state, the action that was performed was a good action and is updated in A-table.

If,

$$v_{s_i}^{g}(t+1) > v_{s_i}^{g}(t) \tag{5}$$

then

$$A_{s_i}^{u} = a_i \tag{6}$$

Continuous actor critic learning automaton uses state values to update the A-table, therefore it is considered to be the fastest of all the reinforcement learning algorithms. Fig.4 provides a complete flow of CACLA algorithm.

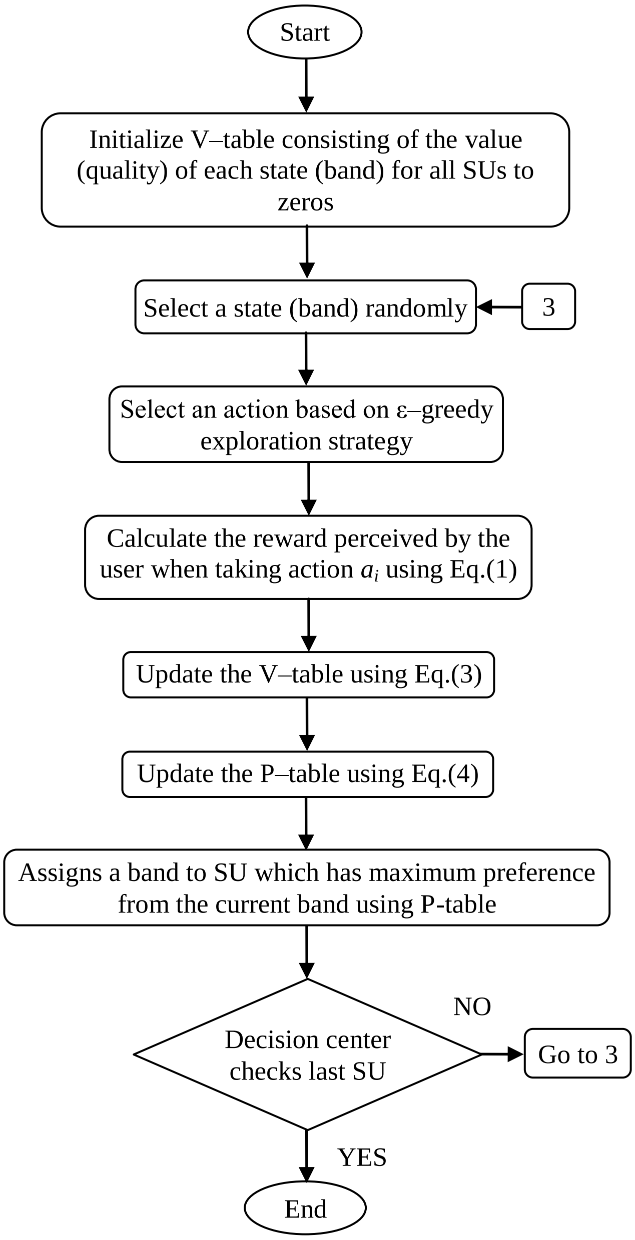
Start
Initialize V–table consisting of the value (quality) of each state (band) for all SUs to zeros
Select a state (band) randomly
3
Select an action based on ε–greedy exploration strategy
Calculate the reward perceived by the user when taking action a_i using Eq.(1)
Update the V–table using Eq.(3)
Update the P–table using Eq.(4)
Assigns a band to SU which has maximum preference from the current band using P-table
Decision center checks last SU
NO
Go to 3
YES
End

Fig.3. Actor Critic flow chart

4. EVALUATION OF CACLA SCHEMES

The CACLA algorithm was simulated using MATLAB v.2010. Table.1 shows the main simulation parameters. The performance of proposed system is compared with the existing Q- learning scheme.

Table.1. Main simulation parameters

Parameter	Value
Number of bands	6
Number of SU	1 - 6
Probability of PU returning to band	0.1 - 0.6
Bandwidth	2kHz – 12kHz
SNR	5dB to 15dB

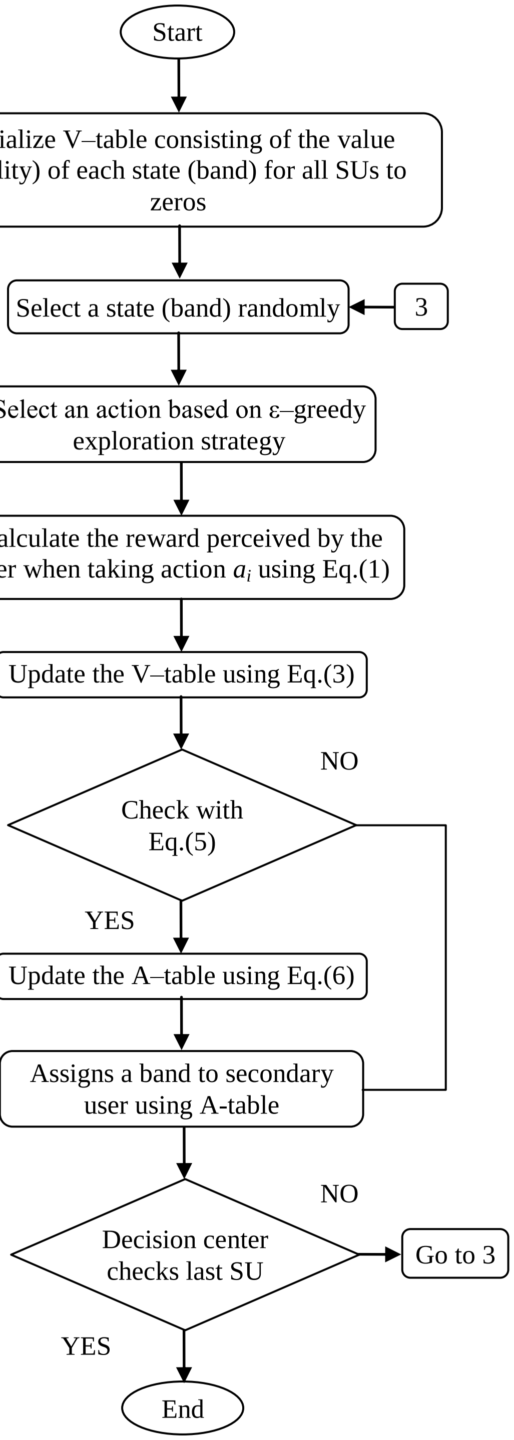
Start
Initialize V–table consisting of the value (quality) of each state (band) for all SUs to zeros
Select a state (band) randomly
3
Select an action based on ε–greedy exploration strategy
Calculate the reward perceived by the user when taking action a_i using Eq.(1)
Update the V–table using Eq.(3)
NO
Check with Eq.(5)
YES
Update the A–table using Eq.(6)
Assigns a band to secondary user using A-table
NO
Decision center checks last SU
Go to 3
YES
End

Fig.4. CACLA flow chart

4.1 EXECUTION TIME

The time taken by SU to switch over from one band to another was observed during simulation. As shown in Fig.5 the execution time of CACLA is lesser than that of Actor–Critic learning and Q–learning schemes. The reason behind this time saving is that in Q–learning the Q–table is updated every time and value table is not used to update the Q-table, but in CALCA and Actor Critic learning schemes the value table is used periodically. Usage of value table speeds up the learning process. In Actor Critic learning scheme value table is used to update the preference table. Therefore, the execution time of CACLA and Actor Critic learning is always lesser than that of Q–learning. The execution of CACLA is still lesser than Actor Critic learning scheme. This is because in CALCA scheme the action table is updated based on value table. First value V- table is updated. If the current V is greater than previous then action table is updated. But this is not the case in Actor Critic learning and Q–learning, because the Q–table and the preference table is updated every time during the learning process. The variance of execution time of Q–learning is less when compared to other two schemes (Fig.6). It is also observed that the variance in execution time of CACLA is higher compared to other two schemes.

4.2 SUM BIT RATE

The bit rate is estimated using Shannon’s channel capacity theorem during simulation process. The sum of bit rate observed in case of Actor Critic learning is merely equivalent to Q–learning, but for the CACLA scheme it is appreciable higher for the larger number of SUs (Fig.7). This is because in Q–learning/ Actor Critic learning first we update the Q–table/ preference table then the band is assigned to the SU based on the updated Q table/preference table which has maximum quality/preference. But in CACLA scheme only the best actions are used to update the action table. Incidentally, as shown in Fig.7, the bit rates achieved initially under all the schemes are similar. It is also noted that when the number of SU increases, the sum of bit rate accomplished under CACLA scheme increases proportionally than other two schemes.

The Fig.8 shows the variance of sum data rate observed for all the schemes. It is higher for CACLA for larger number of SUs compared to other two schemes. In case of Actor Critic learning scheme and Q–learning scheme it is mere equivalent.

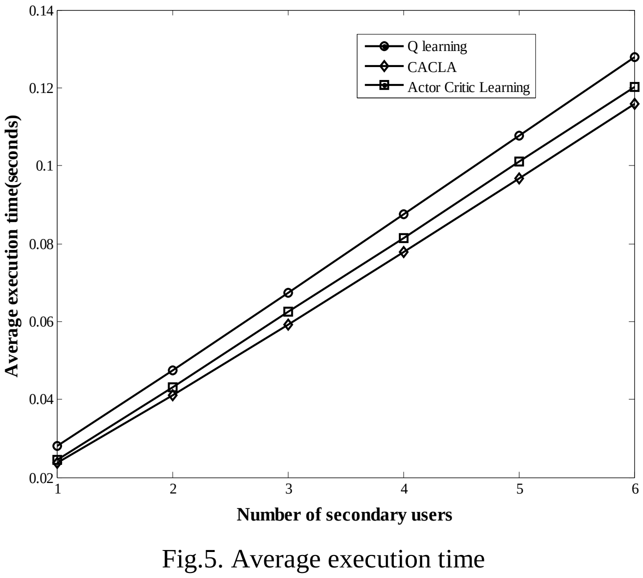

Fig.5. Average execution time

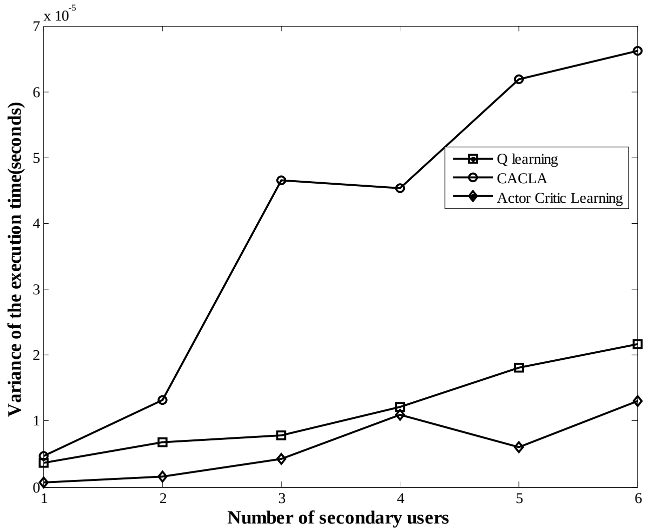

Fig.6. Variance of the execution time

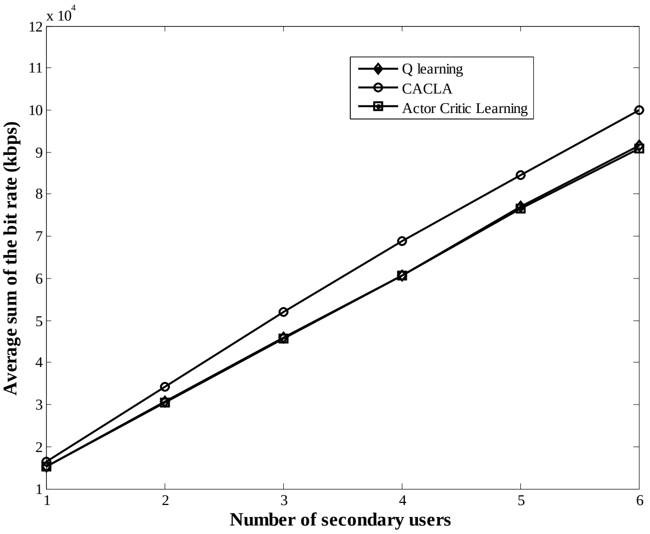

Fig.7. Average sum of bit rate

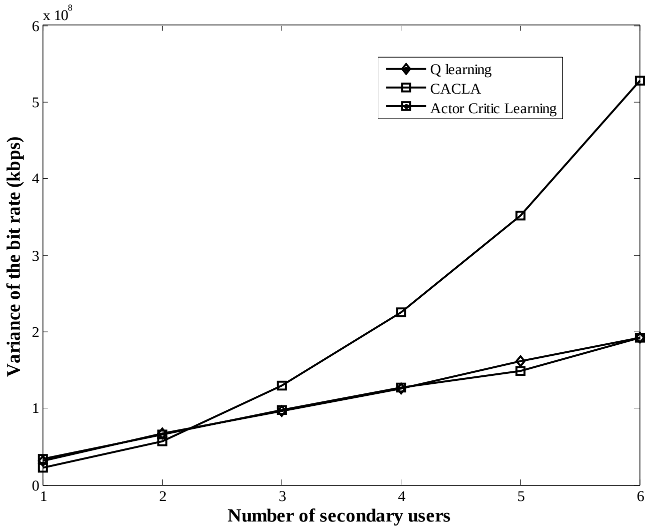

Fig.8. Variance of the bit rate

4.3 AVERAGE THROUGHPUT

The throughput during simulation was observed for each scheme for the incremental number of SUs (Fig.9). Due to the lesser execution time of CALCA compared to other two schemes, the throughput accomplished under CACLA scheme is appreciably higher than Q–learning and Actor Critic learning schemes. Though the bit rate of Actor Critic learning scheme is equivalent as Q–learning, the throughput achieved under Actor Critic learning scheme is slightly higher than Q–learning. This is because execution time of Actor Critic learning scheme is less in comparison with Q–learning scheme.

The Fig.10 shows that the variance of the throughput for Q–learning and Actor Critic learning scheme reduces as the number of users increases but in CACLA the variance is much higher with a certain degree of unpredictability.

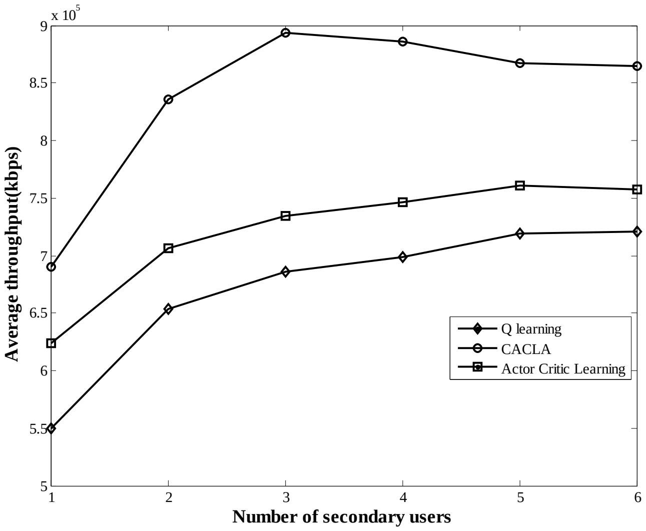

Fig.9. Average throughput

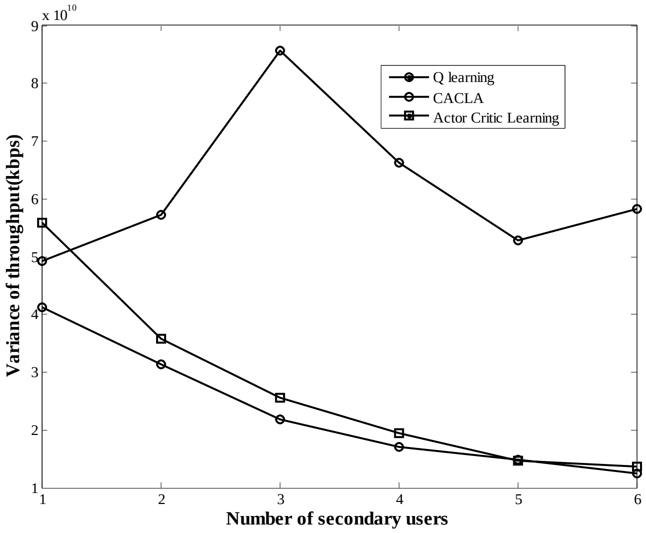

Fig.10. Variance of throughput

5. CONCLUSION

The proposed Continuous Actor Critic Learning Automaton (CACLA) scheme, which is a variation of Actor–Critic learning scheme, performed better than the existing Actor–Critic and Q-learning methods. Although the variance of the observed performance parameters of CACLA were higher, the two main parameters i.e., the sum bit rate and throughput were appreciably high than the other two methods. There is need to incorporate suitable modification in CACLA that reduces variance of the observed performance parameters, which could be the future work.

REFERENCES

[1] US Federal Communications Commission Spectrum Policy Task Force, Report of the Spectrum Efficiency Working Group, ET Docket No. 02-135, 2002.

[2] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications", *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 2, pp. 201-220, 2005.

[3] H. Su and X. Zhang, "Cross-Layer Based Opportunistic MAC protocols for QoS provisioning Over Cognitive Radio Wireless Networks", *IEEE Journal on Selected Areas in Communications*, Vol. 26, No. 1, pp. 118-129, 2008.

[4] Z. Quan, S. Cui and A.H. Sayed, "Optimal Linear Cooperation for Spectrum Sensing in Cognitive Radio Networks", *IEEE Journal of Selected Topics in Signal Processing*, Vol. 2, No. 1, pp. 28-40, 2008.

[5] C.T. Chou, S. Shankar, H. Kim and K.G. Shin, "What and How Much To Gain By Spectrum Agility", *IEEE Journal on Selected Areas in Communications*, Vol. 25, No. 3, pp. 576-588, 2007.

[6] S. Srinivasa and S.A. Jafar, "Cognitive Radio Networks: How Much Spectrum Sharing is Optimal?", *Proceedings of IEEE Global Communications Conference*, pp. 3149-3153, 2007.

[7] Q. Zhao, S. Geirhofer, L. Tong and B.M Sadler, "Opportunistic Spectrum Access via Periodic channel Sensing", *IEEE Transactions on Signal Processing*, Vol. 56, No. 2, pp. 785-796, 2008.

[8] J. Unnikrishnan and V. V. Veeravalli, "Dynamic Spectrum Access with Learning for Cognitive Radio", *Proceedings of 42nd IEEE Asilomar Conference on Signals, Systems and Computers*, pp. 103-107, 2008.

[9] K. Liu and Q. Zhao, "Distributed Learning in Cognitive Radio Networks: Multi-Armed Bandit with Distribute Multiple Players", *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 3010-3013, 2010.

[10] R. S. Sutton and A. G. Brato, "*Reinforcement Learning: An Introduction*", A Bradford Book, The MIT Press, 1998.

[11] PavithraVenkataraman, Bechir Hamdaoui and Mohsen Guizani, "Opportunistic Bandwidth Sharing through Reinforcement Learning", *IEEE Transactions on Vehicular Technology*, Vol. 59, No. 6, pp. 3148–3153, 2010.

[12] Pan Zhou, Yusun Chang and John A. Copeland, "Reinforcement Learning for Repeated Power Control Game in Cognitive Radio Networks", *IEEE Journal on Selected Areas in Communications*, Vol. 40, No. 1, pp. 54–69, 2012.

[13] Animashree Anandkumar, Nithin Michael, Ao Kevin Tan and Ananthram Swami, "Distributed Algorithms for Learning and Cognitive Medium Access with Logarithmic Regret", *IEEE Journal on Selected Areas in Communications*, Vol. 29, No. 4, pp. 731-745, 2013.

[14] Francisco Bernado, Ramon Agusti, Jordi Perez- Romero and Oriel Sallent, "An Application of Reinforcement Learning for Efficient Spectrum Usage in Next-Generation Mobile Cellular Networks", *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, Vol. 40, No. 4, pp. 477-484, 2010.

[15] Ivo Grondman, Lucian Busoniu, Gabriel A. D. Lopes and Robert Babuska, "A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients", *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, Vol. 42, No. 6, pp. 1291 - 1307, 2012.

[16] Pavithra Venkatraman, "Opportunistic Bandwidth Sharing through Reinforcement Learning", M.S. Thesis, Department of Electrical and Computer Engineering, Oregon State University, 2010.