

# Cognitive Spectrum Decision via Machine Learning in CRN

A.M. Koushik, Fei Hu, Ji Qi and Sunil Kumar

**Abstract** In this research, we propose cognitive spectrum decision model comprised of spectrum adaptation (via Raptor codes) and spectrum handoff (via transfer learning) in Cognitive Radio Networks(CRN), in order to enhance the spectrum efficiency in multimedia communications. Raptor code enables the Secondary User (SU) to adapt to the dynamic channel conditions and maintain the Quality of Service (QoS) by prioritizing the data packets and learning the distribution of symbols transmission strategy called *decoding-CDF* through the history of symbol transmissions. Our scheme optimizes the acknowledgement (ACK) reception strategy in multimedia communications, and eventually increases the spectrum decision accuracy and allows the SUs to adapt to the channel variations. Moreover, to enhance spectrum decision in a long term process, we use Transfer Actor Critic Learning (TACT) model to allow the newly joined SU in a network to learn the spectrum decision strategies from historical spectrum decisions of the existing ‘expert’ SUs. Experimental results show that our proposed model works better than the myopic spectrum decision which chooses the spectrum decision actions based on just short-term maximum immediate reward.

**Keywords** Cognitive Radio Network (CRN) · Cognitive Spectrum Decision (CSD) · Raptor codes · Machine learning · Transfer Actor-Critic Learning (TACT)

## 1 Introduction

As per the Federal Communication Commission (FCC) the current frequency spectrum is under-utilized [1]. To utilize the vacant spectrum called as *Back-hauls*, the

---

A.M. Koushik(✉) · F. Hu · J. Qi  
Electrical and Computer Engineering, University of Alabama, Tuscaloosa, AL, USA  
e-mail: koushik@crimson.ua.edu

S. Kumar  
Electrical and Computer Engineering, SDSU, San Diego, CA, USA

© Springer International Publishing Switzerland 2016  
S. Latifi (ed.), *Information Technology New Generations*,  
Advances in Intelligent Systems and Computing 448,  
DOI: 10.1007/978-3-319-32467-8\_2

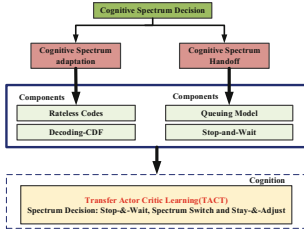
concept of Cognitive Radio Networks (CRNs) has been brought up in which the unlicensed Secondary Users (SUs) use the idle spectrum when the licensed Primary Users (PUs) are not using it. But due to the random arrival of PUs, the SUs cannot utilize the idle spectrum efficiently.

In this research we target spectrum decision (SD) problem, which is to manage the SU transmission behaviors in order to adopt to the dynamic CRN spectrum conditions. Especially we focus on two sub-issues in SD (see Fig. 1): The first one is spectrum adaptation, that is, how a SU adjusts its transmission behaviors (such as packet sending pace) under highly dynamic channel conditions. The second one is spectrum handoff, which requires a SU switches to a new channel if the PU comes back.

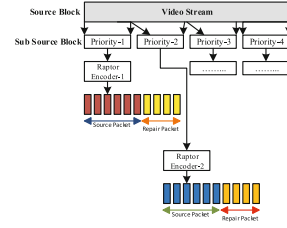
On the issue of spectrum adaptation, it is difficult to adjust the sending rates based on the immediate observation of the channel conditions that could change so quickly. To overcome such an issue, we will use raptor codes [2] to control the sending pace. The raptor code is a rateless and regret free, i.e. the lost packet can be recovered using the earlier symbols received, and it makes use of all the symbols received for packet recovery (i.e., no symbols are wasted). To further improve spectrum efficiency, we employ *decoding-CDF* called as ratemore protocol [3] to transmit the sufficient amount of symbols in a dynamic and time varying channel. Without decoding CDF, the sender may send many redundant symbols when the channel condition is good which consumes more bandwidth unnecessarily or the symbols sent may not be enough to decode the packet successfully in a poor channel condition.

For spectrum handoff, we propose to use machine learning algorithms to manage the channel switching under dynamic channel conditions. Especially we adopt node-to-node transfer learning for spectrum handoff control, (i.e., a node learns from another node on how to switch to a new channel), instead of using self-learning, (i.e., a node learns the channel switching control by itself). A new joined SU can learn from a neighboring SU (called "expert") that has similar radio conditions as itself. A benefit of using node-to-node knowledge transfer is that we can reduce the time taken to converge to optimal value in a global optimization algorithm. Particularly, we employ Transfer Actor Critic Learning (TACT) [4] to allow a "student" SU to make correct spectrum decision by learning from an "expert" SU, which shares the optimal policy acquired from its historical spectrum decision with the learning SU. One more advantage of TACT is that, as the learning progresses, the student SU can start adapting to the CRN environment by its own. This is beneficial since each SU may experience different, dynamic channel conditions due to the independent and random mobility of each SU. Figure 1 shows our cognitive spectrum decision model.

The rest of this paper is organized as follows: Section 2 briefly summarizes the related work. Section 3 then details our CDF and TACT based spectrum decision model. The simulation results are given in Section 4, and Section 5 concludes this paper.



**Fig. 1** The concept of Cognitive Spectrum Decision



**Fig. 2** Prioritized Raptor codes.

## 2 Related Work

**Teaching Based Spectrum Decision Enhancement:** Teaching based machine learning algorithms for wireless protocol design have been proposed in Apprenticeship learning (AL) [5] and Docitive learning models [6]. Docitive learning [6] have been used for interference management in femtocells. But those models cannot provide the explicit channel selection parameters. They also do not describe how to search an expert node. Besides, in AL model [5], a student node learns from the expert node all the time, and the student node should encounter exactly the same channel conditions as the expert node does during the entire learning process. In our TACT-based approach, a SU learns from expert at the initial stage of transmission and later learns on its own.

**Rateless Codes with Decoding CDF for Enhanced Wireless Transmission:** Rateless codes enable to decode the packets with low Packet error rate in wireless transmissions. Rateless codes in CRN have been proposed in [7] in which SUs act as relay nodes to forward PUs data packets. Whereas in our approach we employ rateless codes with decoding CDF to optimize symbol transmission strategy adjust to the time varying channel condition adaptively.

## 3 Cognitive Spectrum Decision

### 3.1 Spectrum Adaptation Based on CDF-Enhanced Raptor Codes

Each SU selects a channel for communication by considering various parameters such as channel holding time, channel sensing accuracy, Packet dropping rate, etc. The reason of considering all those parameters is due to the dynamic nature of the CRN channels that exhibit time-varying link quality. Poor link quality can induce high packet loss rate. To reduce the packet loss rate the sender needs to adjust the packet

transmission rate based on channel conditions. As an example, in 802.11 links, the sender determines the sending rate by measuring link signal-to-noise (SNR) ratio and uses it to estimate the constellation points associated with the respective modulation modes. It is hard to achieve smooth rate adaption since we typically have only a few pre-determined rates available in the system. Therefore, it is challenging to design a dynamic spectrum adaptation scheme in CRNs since channel conditions vary so quickly, even within the very short individual packet transmission time.

Rateless codes are also called as *regret-free* codes, which treat all transmitted symbols equally and no symbol is discarded since any symbol can be used to decode the later received symbols. Rateless codes have shown promising performance in multimedia over CRNs. In addition, in the sender side, each packet is disintegrated into symbols with added small redundancy. This enables the receiver to decode the symbols successfully as long as it has enough symbols to decode. Moreover, the sender does not need to make changes in its modulation and encoding schemes. In other words, it is "rateless" since the sending rates do not need to be micro-adjusted based on the channel conditions. The sender simply keeps sending symbols until the receiver is able to decode all the packets, and then the sender sends the next window of packets. For a well-designed rateless codes, the number of symbols transmitted closely tracks the variation in channel conditions.

Raptor codes assign different priorities to different packets, and the packets with higher priority are given more redundancy of symbols. From Fig. 2 we can see that, initially the packet is decomposed into pieces (group of symbols). Those pieces first pass through outer encoder called as LDPC codes, and they then pass through inner encoder called as LT codes. Hence, encoding can be characterized by  $(K, C, \theta(x))$ , here  $K$  means the number of message blocks (pieces),  $C$  is the outer code result (with block size  $L$ ). Thus we have  $L$  intermediate symbols after passing outer encoder. The last  $L - K$  symbols are redundant symbols.  $\theta(x)$  is the degree distribution of LT codes. The  $L$  intermediate symbols are encoded with LT code to generate  $N$  encoded symbols. In total  $N$  symbols are transmitted over the lossy wireless channel. Even some packets are received with errors, it can be successfully decoded by using the previously received symbols. Let  $N_r$  be the number of received encoded symbols. The decoding failure probability,  $P_e(\xi_r)$ , is very low. Here  $\xi_r = Nr - K$  is the encoding overhead (i.e., redundancy level) of Raptor codes. Then we have:

$$Pe(\xi_r) = 0.85 \times 0.567^{\xi_r} \quad (1)$$

The average communication overhead, induced due to the extra added symbols among the source symbols is:

$$\rho = \frac{1}{K} \sum_{i=0}^{\infty} (i \cdot (P_e(i-1) - P_e(i))) \approx \frac{2}{K} \quad (2)$$

According to (2), we can see that we just need to transmit approximately 2 extra symbols to decode the transmitted packet successfully because extra added symbols (in average) should be:  $K \times \rho = 2$ . In addition, we can generate more symbols for high priority packets so that the receiver can have ample amount of symbols to decode the high priority packets successfully. Let  $L_1$  denote the highest priority,  $L_2$  the second highest, and so on. And we have  $K_i$  source symbols with priority  $L_i$ . Also  $\xi_r(K_i)$  denotes the number of extra symbols for the priority  $L_i$ . Then the minimum coding overhead induced is  $\rho(K_i)$ , and the percentage of additional symbols among the total source symbols for priority  $L_i$ , should be:

$$\rho(K_i) = \frac{K_i \times PER + \xi_r(K_i)}{(1 - PER) \times K_i} \quad (3)$$

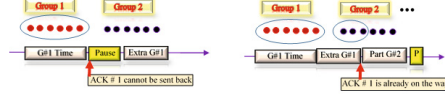
Here PER is the packer error rate. Conventional raptor codes treat all packets equally. Hence, our Unequal Error Protection (UEP) based Raptor codes can adaptively adjust the overhead of raptor codes based on PER for data with different priority levels.

**Decoding CDF for Enhanced Raptor Codes:** Achieving higher throughput in wireless networks is an important goal. In rateless codes, without careful scheduling and estimation the sender will send the data, meaning that in rateless codes the amount of symbols sent may be too redundant when the channel quality is good or the sent symbols may not be sufficient to decode the packet successfully in a lossy channel. Sending too many redundant symbols consumes bandwidth unnecessarily and the bandwidth cannot be used efficiently. On the otherhand, when packets are not being decoded successfully the QoE deteriorates. Hence, both will not help to attain the required QoS and QoE. Hence a link layer protocol called *Ratemo*, is used to obtain the cumulative distribution for the probability of successful decoding of a sent packet. Using such a distribution (called as decoding CDF) the sender can estimate the number of symbols to be in the present channel condition. Using Ratemo protocol [3] we can design a proper spectrum adaptation strategy to estimate how many symbols can be transmitted before pausing for the feedback. In addition, the proper pausing intervals can also be determined to maintains the good QoS without introducing much communication overhead.

$$F(x) = \int_0^x \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \quad (4)$$

Where,  $x$ ,  $\mu$  and  $\sigma$  are the Number of samples(NS), mean and variance estimated out of the symbols transmission history respectively.

The decoding CDF [3] can be learnt online using the history of transmitted symbols. The Gaussian approximation model can be used for CDF learning. We just need to estimate the parameters including Gaussian mean,  $\mu$ , and variance,  $\sigma^2$ . Both can be estimated in exponentially weighted accumulator fashion. Equation (4) shows CDF learning principle by using Gaussian approximation with learning rate  $\alpha \in [0, 1]$ .



**Fig. 3** (left) No CDF ; (right) With CDF.

Figure 3 shows the benefit of using CDF. The receiver cannot send the ACK unless there are ample amount of symbols to decode the G#1 packets successfully (see left figure); but when it has sufficient symbols to decode the packet (see right figure), the receiver sends the feedback. By using those received symbols, the receiver can draw a distribution of packet decoding.

### 3.2 Spectrum Handoff Control Based on TACT Model

The spectrum decision can be modeled as a Markov Decision process (MDP). It can be represented by a tuple as  $(S, A, T, R)$ , where  $S$  depicts the set of system states,  $A$  is the set of system actions at each state.  $T$  represents the transition probability, where  $T = P(s, a, s')$  is the probability of transition from state  $s$  to  $s'$  when the action  $a$  is taken, and  $R : S \times A \mapsto R$  is the reward or cost function, which depicts the reward for taking an action  $a \in A$  in state  $s \in S$ . In MDP we intend to find the optimal policy  $\pi^*(s) \in A$ , i.e. a series of actions  $\{a_1, a_2, a_3, \dots\}$  for state  $s$ , in order to maximize the total discount reward function.

**States,  $S$  :** For SU- $i$ , the network state at  $(j + 1)th$  channel assignment stage is  $s_{ij} = \{\chi_{ij}^{(k)}, \xi_{ij}^{(k)}, \rho_{ij}^{(k)}, \phi_{ij}^{(k)}\}$ . Where  $k$  is the channel being used.  $\chi_{ij}^{(k)}$  is the idle or busy status of the channel.  $\xi_{ij}^{(k)}$  is the channel quality determined using Packet Error Rate(PER).  $\rho_{ij}^{(k)}$  is the channel traffic load condition determined using non-preemptive M/G/1 queueing model[5][4] and  $\phi_{ij}^{(k)}$  denotes priority.

**Actions,  $A$  :** Three actions are considered for iSM scheme: 1. stay-and-await: stay in the same channel and hold the traffic until the channel condition is above a pre-set threshold; 2. stay-and-adjust (transmit more or less symbols) until the stable reward, Mean Opinion Score(MOS) value is met using decoding-CDF, and 3. spectrum handoff: switch to a new channel; We denote  $a_{ij} = \{\beta_{ij}^{(k)}\} \in A$  as the candidate of actions for  $SU_i$  on state  $s_{ij}$  after the assignment of  $(j + 1)th$  channel.  $\beta_{ij}^{(k)}$  represents the probability of choosing action  $a_{ij}$ . A particular action is selected based on softmax policy as below,

$$\pi^{(k)}(s^{(k)}, a) = \frac{\exp(\frac{Q(s,a)}{\tau})}{\sum_{a' \in A} \exp(\frac{Q(s,a')}{\tau})} \quad (5)$$

Where  $\tau$  is called temperature. The high temperature indicates the exploration of the unknown state-action pairs.

**Reward :** We adopt Mean opinion score (MOS) as the reward function using the following equation:

$$MOS = \frac{a_1 + a_2 FR + a_3 \ln(SBR)}{1 + a_4 PER + a_5 (PER)^2} \quad (6)$$

Where FR is frame rate, SBR is Sender Bit Rate, and TPER is Total Packet Error Rate. Here  $a_1, a_2, a_3, a_4$ , and  $a_5$  are coefficients estimated using regression analysis.

**Self-Learning via Q-Learning:** When the new SU (denoted as SU-i) joins the network and if there is no expert SU that has similar QoS requirement to itself, then it can learn spectrum decision actions by itself via Q-learning algorithm, as shown in Fig. 4. Q-learning aims to find the optimal action to maximize the MOS at the current policy  $\pi^*(s_{i,j}, a_{i,j})$  in the process of  $(j + 1)th$  channel assignment to SU-i. The fairness of the action,  $a_i$ , taken in state  $s_i$  given policy  $\pi$ , can be found by action-value function,  $Q^\pi(s, a)$ , which is given by

$$Q^*(s, a) = E(R_{i,j+1}) + \gamma \sum_{s'} P_{s,s'}(a) \max_{a' \in A} Q^*(s', a'), \quad \gamma \in (0, 1) \quad (7)$$

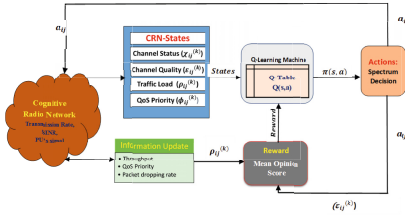


Fig. 4 Q-learning based CSH

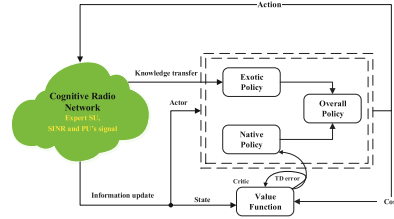


Fig. 5 TACT based CSH.

**TACT-Based Spectrum Decision:** Q-learning could take a long time to converge to an optimal solution due to the difficulty of selecting suitable state and initial parameters in the Markov model. To increase the efficiency of spectrum decision, the learning process has to be made fast. We thus propose to use Transfer Actor Critic Learning (TACT) to enhance the learning process. In TACT, the newly joined SU finds an expert SU which has similar QoS requirement as itself. For example, a SU with video transmission could find an expert SU with video applications instead of a SU with data transmission tasks. The expert SU exchanges the optimal policy learnt during its previous spectrum decisions with the student SU. Moreover, if the SU cannot learn from the expert anymore due to radio condition mismatch, the SU can learn from its own historical spectrum decision records.

TACT comprises of three components: *actor*, *critic*, and *environment*. For a given state, the actor selects and executes an action using softmax policy. This causes a

transition from one state to another with certain reward that is fed back to actor. Then the critic calculates the *time difference* (TD error) to evaluate the action taken and updates the value function. After receiving the feedback from the critic, the actor updates the policy. Figure 5 depicts the TACT principle.

**1. Transferring the Knowledge:** Both the policy and value function are updated separately. This makes the transfer of policy knowledge easier than Q-learning model.

**i. Action Selection:** Initially at state  $s_{ij}$  the node is using channel  $k$ , and  $SU - i$  chooses an action using eqn(6) to find an optimal policy in a current channel.

**ii. TD error calculation and State-value function update:** TD time difference can be calculated as,

$$\delta(s, a) = R_{s,a} + \gamma \sum_{s' \in S} P(s'|s, a) V(s') - V(s) = R_{s,a} + \gamma V(s') - V(s)$$

Where,  $R(\cdot)$  is the reward, and  $V(\cdot)$  is the state value function. Subsequently, the state-value function can be updated as

$$V(s') = V(s) + \alpha(v_1(s, m))\delta(s, a) \quad (8)$$

Where  $v_1(s, m)$  indicates the occurrence time of the state  $s$  in these  $m$  stages.  $\alpha(\cdot)$  is a positive step-size parameter that affects the convergence rate.

**iii. Policy Update:** The policy is updated using the feedback from the critic as follows,

$$p(s, a) = p(s, a) - \beta(v_2(s, a, m))\delta(s, a) \quad (9)$$

Where  $v_2(s, a, m)$  denotes the occurrence time of action,  $a$  at state  $s$  in these  $m$  stages.  $\beta(\cdot)$  denotes the positive step size parameter.

**2. Overall Policy Update:** The overall spectrum decision policy compares the policy of the expert with that of the student  $SU$ , as shown in (10). The overall policy is determined by the combination of the native policy  $p_n$  and an exotic policy (expert policy)  $p_e$ . Assume at channel  $k$ , the state is  $s^{(k)}$  and the chosen action is  $a^{(k)}$ . Accordingly, the overall policy can be updated as [4]

$$p_o^{(k+1)}(s, a) = [(1 - \omega(v_2(s, a)))p_n^{(k+1)}(s, a) + \omega(v_2(s, a))p_e^{(k+1)}(s, a)]_{-p_t}^{p_t} \quad (10)$$

Where  $[x]_m^n$  with  $n > m$ , indicates the Euclidean distance of interval  $[m, n]$ , i.e.  $[x]_m^n = \min(x, m)$ ;  $[x]_m^n = n$  if  $x > n$ ; and  $[x]_m^n = x$  if  $m \leq x \leq n$ . In this scenario,  $m = -p_t$  and  $n = p_t$ .  $p_o^{(m+1)}(s, a) = p_o^{(m)}(s, a)$ ,  $\forall a \in A$  but  $a \neq a_{ij}$ . Apart from that,  $p_n(s, a)$  also updates itself according to (10). Interestingly, at the beginning of the process, the exotic policy  $p_e(s, a)$  dominates the overall learning policy, i.e., the learning  $SU-i$  takes the action which is optimal for expert  $SU$ ; and as the learning progresses, the effect of exotic policy starts decreasing with parameter  $\omega \in (0, 1)$ . The transfer rate follows  $\omega \mapsto 0$  as the number of iterations goes to  $\infty$ . This makes



student SU-i get rid of negative guidelines from expert SU and can learn by itself according to new channel conditions.

## 4 Simulation Results

In our simulation, we assume that the selected channel has high idle duration, channel condition is generally good (i.e., low packet error rate), and the network congestion (Packet dropping rate) is not serious. In addition, we assume the video transmission has highest priority and its waiting time in queue is very low.

### 4.1 Spectrum Adaptation Based on Decoding CDF

In this section, we evaluate the decoding CDF distribution and learning performance after utilizing raptor codes. Figure 6 shows the decoding CDF learning result using Algorithm 1 with learning rate specified in Table 1 for different SRN values. i.e., from -5 dB to 25 dB. The graph clearly depicts that, with high SNR (say 25 dB), less symbols (100 from Fig. 6) have to be transmitted in order to decode the packet successfully; whereas at low SNR (-5 dB) large number of symbols (>2300) decode the transmitted packet successfully.

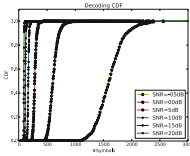


Fig. 6 decoding CDF

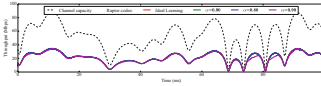


Fig. 7 Channel throughput.

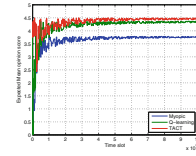


Fig. 8 Learning Performance.

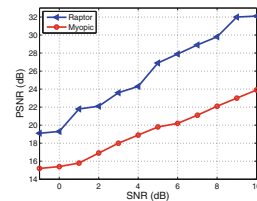
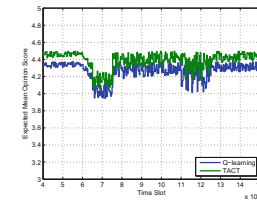
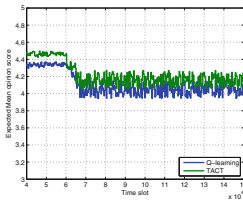
In Fig. 7, we estimate the throughput that can be achieved using the decoding CDF strategy and pausing intervals in raptor codes over a Rayleigh fading channel in the duration of 100ms. In addition, it is assumed that the SU is moving at a speed of 10m/s with average link SNR=15dB. The raptor code has the throughput almost half the Shannon capacity due to the fact that it needs to transmit some *empty* packets (no payload) in order to meet the latency requirements.

### 4.2 TACT Enhanced Spectrum Decision

In this section we examine the performance of our TACT based spectrum decision with the Q-learning and Myopic spectrum decisions. We assume that the peak sending

rate of each SU is 3Mbps. Assume that all SUs adopt rateless codes, and the expert SU teaches a learning SU about its spectrum decision policy as determined in section III-B. We examine 3 cases. 1). Fast moving SU, 2). Learning without decoding CDF, and 3) learning with decoding CDF. Myopic spectrum decision considers only the immediate maximum reward without considering the long-term effect. Whereas Q-learning scheme is a self-learning scheme which learns about the spectrum decision from the scratch. It eventually takes longer time to converge than TACT-based scheme. TACT learns from the expert SU to reduce the learning time and to get adapted to the dynamic channel conditions quickly in order to enhance the spectrum efficiency. We consider MOS as the reward since at the receiver it is hard to measure the video PSNR (peak SNR). Our scheme has been verified in Fig. 8, in which the reward, MOS (from (8)) achieved in TACT is high compared to Q-learning and myopic. TACT learns from the expert SUs policy. Thus it takes less time to achieve optimality. Q-learning takes the action with the best performance in the future but takes initial time to converge, whereas myopic scheme takes the action only considering the immediate reward.

Figure 9 depicts the learning performance when the SU is moving fast and does not use Ratemore protocol. The SU experiences variation in channel conditions frequently due to channel fading, signal attenuation and small coherence time, etc. Those factors eventually affect the reward value as shown in Fig. 9. If the Ratemore protocol (decoding CDF) is not employed, the SU cannot recover from the low reward and it continues with the same lower value. If the decoding CDF is employed, the SU can pause properly for the feedback from the receiver, and it eventually learns and executes the strategy, i.e., transmitting enough symbols for the receiver to decode the packet successfully. This conclusion is shown in the Fig. 10 in which the SU is able to recover its optimal value with the aid of the strategy learnt from decoding CDF.



**Fig. 9** SU Without decoding CDF **Fig. 10** SU with decoding CDF **Fig. 11** PSNR v/s SNR.

Figure 11 shows the variation of PSNR with respect to different SNRs for the video frames shown in Fig. 12. Compared to general myopic scheme (without using learning model), our method improves the PSNR all the times. Figure 12 shows the video resolution comparisons between our scheme and myopic spectrum decision. We can see our scheme outperforms the myopic one.



**Fig. 12** Video effect comparisons of learning-based and myopic spectrum decision schemes.

## 5 Conclusions

In this paper we have demonstrated our cognitive spectrum decision scheme using rateless codes, decoding CDF, and machine learning algorithm called TACT. It is observed that the rateless codes along with decoding CDF can maintain the QoS over dynamic channel conditions and optimize the feedback and symbol transmission strategies. TACT-based learning algorithm enhances the process of adaptation to the channel conditions. This increases the throughput and spectrum utilization. Our cognitive spectrum decision can be applied in multimedia communications over CRNs.

## References

1. Commission, F.C., et al.: Spectrum Policy Task Force Report, FCC 02-155 (2002)
2. Shokrollahi, A.: Raptor codes. *IEEE Transactions on Information Theory* **52**(6), 2551–2567 (2006)
3. Iannucci, P.A., Perry, J., Balakrishnan, H., Shah, D.: No symbol left behind: a link-layer protocol for rateless codes. In: *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, pp. 17–28. ACM (2012)
4. Hu, F., Koushik, A.M.: Intelligent Spectrum Mobility based on Transfer Actor-Critic Learning for Rateless Transmissions in Cognitive Radio Networks. *IEEE Transactions on Mobile Computing*, Submitted (2015)
5. Wu, Y., Hu, F., Kumar, S., Sun, Q.: Apprenticeship Learning based Spectrum Decision in Multi-Channel Wireless Mesh Networks with Multi-Beam Antennas. *IEEE Transactions on Mobile Computing*, Submitted (2015)
6. Giupponi, L., Galindo-Serrano, A., Blasco, P., Dohler, M.: Dognitive networks: an emerging paradigm for dynamic spectrum management [dynamic spectrum management]. *IEEE on Wireless Communications* **17**(4), 47–54 (2010)
7. Zhi, X.-Y., He, Z.-Q., Wu, W.-L.: A Novel Cooperation Strategy Based on Rateless Coding in Cognitive Radio Network. *IJACT: International Journal of Advancements in Computing Technology* **4**(8), 333–347 (2012)



<http://www.springer.com/978-3-319-32466-1>

Information Technology: New Generations  
13th International Conference on Information  
Technology

Latifi, S. (Ed.)

2016, XXI, 1306 p. 495 illus., Softcover

ISBN: 978-3-319-32466-1