

The Implementation of Spectrum Sensing and Spectrum Allocation on Cognitive Radio

Pitcha Rungsawang, Amnach Khawne

Department of Computer Engineering, Faculty of Engineering,
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
pitcha.rsw@gmail.com, amnach.kh@kmitl.ac.th

Abstract— *Cognitive Radio is a system designed to use radio spectrum in the most efficient way possible—as demands for radio spectrum have been heavily increased nowadays. The system works by allowing unlicensed users to use radio channel on the licensed users' spectrum without interfering with the licensed users. The system needs spectrum sensing to verify the usage of each specific spectrum and in ever-shifting environment, Reinforcement Learning is also used for establishing spectrum allocation as well.*

Keywords— Cognitive Radio, Spectrum Sensing, Spectrum Allocation, Reinforcement Learning, USRP, LabView

I. Introduction

Radio Spectrum is a primary form of wireless communication. In order to reserve radio frequency, the formal request need to be accepted by a national organization which oversee the management of radio frequency. However, since there are massive usage from countless users and active equipment, the management and distribution of radio frequency are usually insufficient. So the new method of radio frequency management is needed to offer more efficiency and flexibility: Cognitive Radio (CR) [1].

Cognitive Radio allows the unlicensed secondary users (SU) to use the frequency without affecting the licensed Primary Users (PU) by granting usage to the SU while the frequency is not in use by the PU—using the spectrum sensing [2] method to detect usage and look for spectrum hole in the environment. Nevertheless, the difficulty of this method is detecting the user's signal within noise-heavy environment. Speed and accuracy are also additional concerns.

Reinforcement Learning is the most suitable method for the ever-shifting environment; using mathematical calculation to find the best possible spot. Even after arriving at the answer, the system will still take surrounding environment into calculation to find the best long-term answer.

The goal of this experiment is to learn and develop the capability of Cognitive Radio by using Reinforcement Learning. The experiment is conducted on the vacant radio frequency slot, using Universal Software Radio Peripheral (USRP) [3] equipment to study radio frequency communication, along with LabView as a program to generate

environment simulation and assemble .m files to use in MATLAB.

II. Literature Review

Cognitive Radio[1] was introduced by Joseph Mitola and Gerald Q. Maguire in 1999, and keep on developing since then: using Machine-Learning[5] and artificial intelligence[4] to improve the system's performance. Reinforcement Learning is also the subsection of Machine-Learning, suitable for learning about environment surrounding the agent. The process is based on trial and error to finding the best possible state; giving reward when choosing good states and punishment when choosing bad ones[6]. All of those rewards and punishments will then be send back to the agent in order to estimate a pattern which can score the highest—finding the balance between exploration and exploitation. The advantage of this algorithm is even if the best answer is found; the system will continue to learn from available data to try to generate even better answer—causing the algorithm to improve and balance itself out[7].

The aforementioned reward-punishment process is called Weighting Factors[8]. The values of the whole factors are shown in TABLE I. Based on the degree of success, either a reward or a punishment is assigned to the weight of the used spectrum.

TABLE 1. Weighting Factors [8]

Patterns	Before		After	
	Reward	Punishment	Reward	Punishment
Light	1	1	1	-1
Mild	1	0	1	0
Heavy	1	0.5	1	0

The main difference between these patterns is the values assigned to punishment factors. Under normal circumstances, the absolute values of the reward value and the punishment value are equal—hence the 'Light Penalty' pattern. If the attempt for communication fails, however, the weight is directly reduced to zero. Therefore we call it the 'Mild Penalty'. If the user picks up a channel randomly and keeps

using it as long as the quality of communication in that channel is above the requirement, weights will be reduced by 50% immediately. We refer to this pattern as the 'Heavy Penalty' pattern.

There are many past studies dedicated to improving the efficiency of Cognitive Radio, including communication simulation on MATLAB and communication experiment in real time. M. A. K. Abdulsattar and Z. A. Hussein had conducted Energy Detector experiment in real time[9], using TMS320C6713 as digital processor. N. Sharma, D. B. Rawat[10]. also conducted Dynamic Spectrum Access in Cognitive Radio using USRP and LabView.

III. Spectrum Sensing

Spectrum Sensing is mainly used for detecting the usage of primary user on specific frequency. There are many different methods available for the process: Energy Detection, Cyclostationary Feature Detection, Matched Filter Detection, etc. This experiment, however, focuses on Energy Detection. The main drawback of this process is its low efficiency, but it compensates in simplicity; it can be conducted even without prior knowledge of incoming signals. This is the reason for many studies conducted to find a way to improve its efficiency.

The process of detection is as followed:

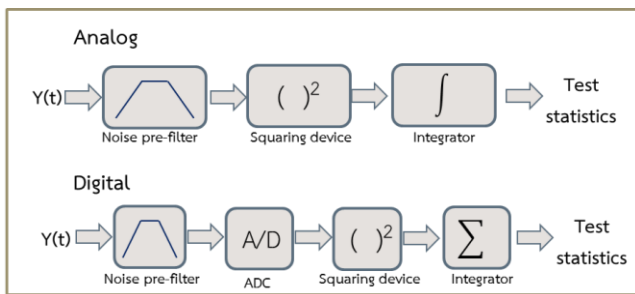


Figure 1. Process of Energy Detection

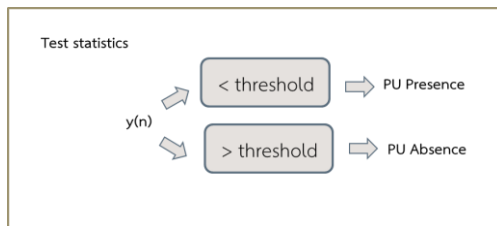


Figure 2. Test statistics

When the system received the input signal $y(t)$, it will generate the sampling by randomize all signals in and send it along as $y(n)$. Then finding the energy by times the amplitude by two and divide by the summary of all amplitudes. The resulting energy will then be compared to the threshold from [11]. If the energy pass the threshold, it means there is a user currently occupied the frequency. If not, then the frequency is vacant.

Threshold used for this experiment is as followed:

$$\lambda_k^* = \sqrt{2} \operatorname{erfc}^{-1}(2P_{fk}), k \neq i$$

There are two hypotheses following the experiment:

$$T_{ED} = \begin{cases} \text{Normal}(\sigma_w^2, \frac{2\sigma_w^4}{N_{ED}}) & ; H_0 \\ \text{Normal}(\sigma_w^2 + \sigma_s^2, \frac{2(\sigma_w^2 + \sigma_s^2)^2}{N_{ED}}) & ; H_1 \end{cases}$$

Normal is Normal Distribution, σ_s^2 is Signal Variation and σ_w^2 is Noise Variation

The efficiency of Spectrum Sensing depends on Probability of Detection (P_d) and Probability of False Alarm (P_f) which can be estimated using these equations:

$$P_d = Q\left(\frac{\lambda_{ED} - (\sigma_w^2 + \sigma_s^2)}{\sqrt{\frac{2}{N_{ED}}(\sigma_w^2 + \sigma_s^2)}}\right)$$

$$P_f = Q\left(\frac{\lambda_{ED} - \sigma_w^2}{\sqrt{\frac{2}{N_{ED}}\sigma_w^2}}\right)$$

λ_{ED} is the Threshold of ED and $Q(\cdot)$ is the Gaussian distribution

IV. Experiment on USRP and LabView

We conducted this experiment by using NI USRP-2920, which is hardware to be used with computer to imitate the signal process of radio. Alongside computers with MATLAB and Labview installed.



Figure 3. USRP 2920

Labview is a program which includes many virtual instruments for research and development alongside USRP or creates simulation by itself. Like programming, users can create and design their own simulation with plenty of tools at their disposal: operator, gate, if condition, while loop, for loop, and virtual instruments. It can also be used to analyze incoming signal, simulate signal from different modulations, and visualize embedded control systems.

We used LabView with USRP to send both signal and noise in real time for USRP to do spectrum sensing with energy detection. The result was unsatisfactory, since energy detection is inefficient within a noise-heavy environment with

low SNR—according to [12]. So we used Q-learning to analyze incoming signal and adjust gain on the radio by using LabView to control USRP.

Theoretically, Q-learning is a trial and error process. When SNR is low, we would adjust gain until we receive answer or acceptable state, but Q-learning will not stop there. It will continue to explore; using randomization to find better answer—making it suitable for spectrum sensing in ever shifting and unpredictable environment, as in the following formula:

$$Q(s,a) = r_{(s,a)} + \gamma \argmax(Q(s',a'))$$

$Q(s,a)$ is a matrix function following:

$$Q(s,a) = \text{zeros}(\text{length}(\text{state}), \text{length}(\text{action}))$$

r is intermediate reward and γ is discount factor of learning and $Q(s',a')$ is a matrix function following:

$$Q(s',a') = \text{zeros}(\text{len}(\text{new state}), \text{len}(\text{new action}))$$

We used weight adjustment setting from table 1 in chapter 2. Learning rate, discount factor, and exploration probability were used with the default setting of 0.5.

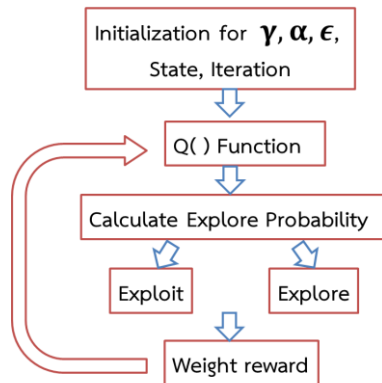


Figure 4. Q-learning Flowchart

The communication between USRP and experimental computer was conducted via Ethernet. We set the IP address of USRP as 192.168.1.2 255.255.255.0 and IP address of computer as 192.168.1.3 255.255.255.0.

We inputted the following parameter into USRP: specific frequency, USRP's IP address, Antenna, Signal Amplification Rate, and Sampling Rate. From there, we generated the Signal to Noise Ratio (SNR) and calculated with the Sampling signal using energy detection, estimated the result with P_d and P_f , and put the findings into graphs.

Below are block diagrams from LabView:

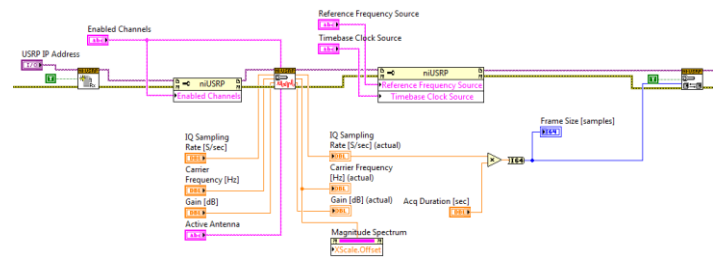


Figure 5. USRP Open Rx Session on Configure Functions

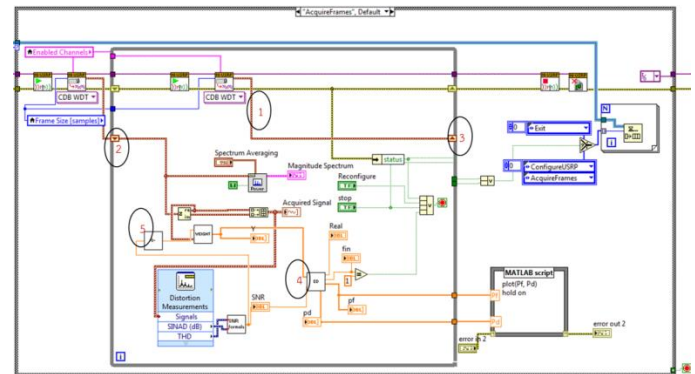


Figure 6. USRP read, write, process signal, energy detector and using SNR to weight for Q-learning

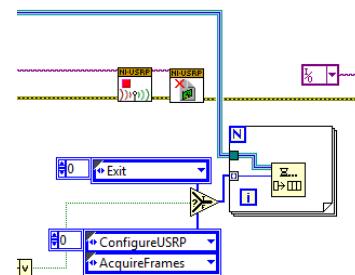


Figure 7. USRP Close Session and abort

Opening Rx session of USRP is showed in figure 5. We configured IP address of USRP, antenna number, sampling rate, etc. It's initializing for radio first. Then we obtain discrete signal. We calculated SNR to energy detection before use it into reward function of Q-learning that we designed in figure 6. Figure 7 we terminated session of USRP at end of program.

From figure 6, point 1, 2, and 3 are complex waveform ($y(n)$) data lines, which we would use in a formula. The process is meant to create waveform of time domain and frequency domain, as well as using $y(n)$ to find SNR in energy detection. Point 4 is a MATLAB script which used SNR and $y(n)$ to calculate P_d and P_f from the theory in chapter 2. Point 5 is the result of using SNR from each iteration to calculate weight and adjust gain of the radio. After that, we plot a ROC graph between P_f and P_d .

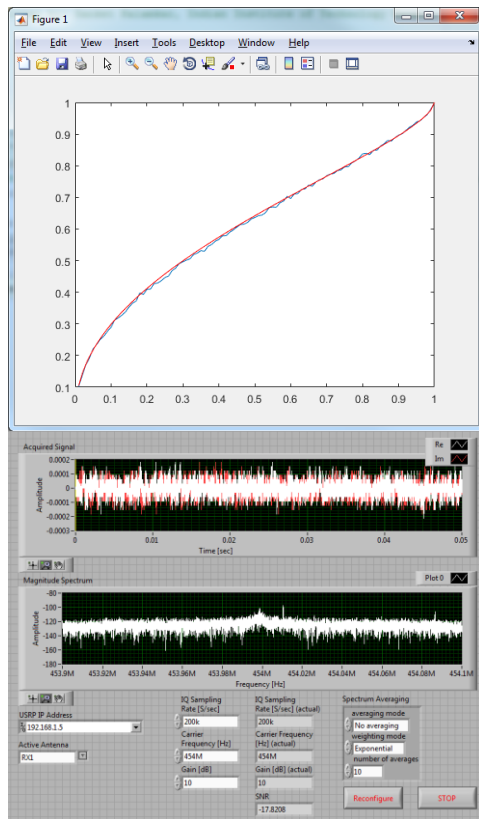


Figure 8. ROC curve between Pd, Pf at SNR = 17.8208 dB

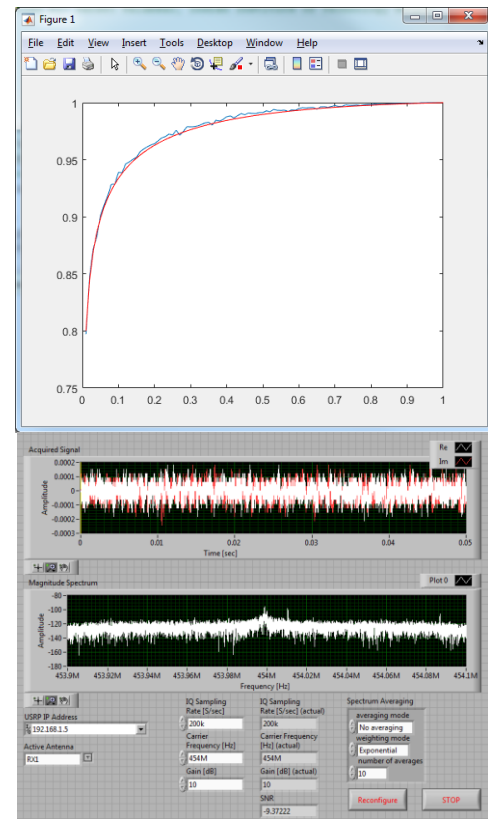


Figure 10. ROC curve between Pd, Pf at SNR = 9.37222 dB

Figure 8 is a result prior to q-learning. Figure 9 and 10 are results after q-learning.

From the result, we can conclude that the outcome of energy detection depends heavily on SNR. When Q-learning is included in the radio, signal detection will become more efficient.

V. Experimental result

The efficiency of spectrum sensing can be compared to Receiver Operator Characteristic (ROC) Curve, generating relation graphs between X as a true positive rate or Sensitivity and Y as false positive rate or 1- Specificity. Therefore, true positive rate of this experiment is probability of detection and false positive rate is probability of false alarm. The best result is the spot on curve farthest from diagonal line. Figure 8 and 9 showed the best result that we had experiment and limit environment.

VI. Conclusion

In this experiment, we conduct spectrum sensing using energy detection method and reward feedback with SNR to weight parameter on the USRP in real time. The efficiency of detection depends largely on the level of noise that real environment. In the future, reinforcement learning will be used alongside energy detection to improve efficiency of Cognitive Radio system for further experiment on USRP.

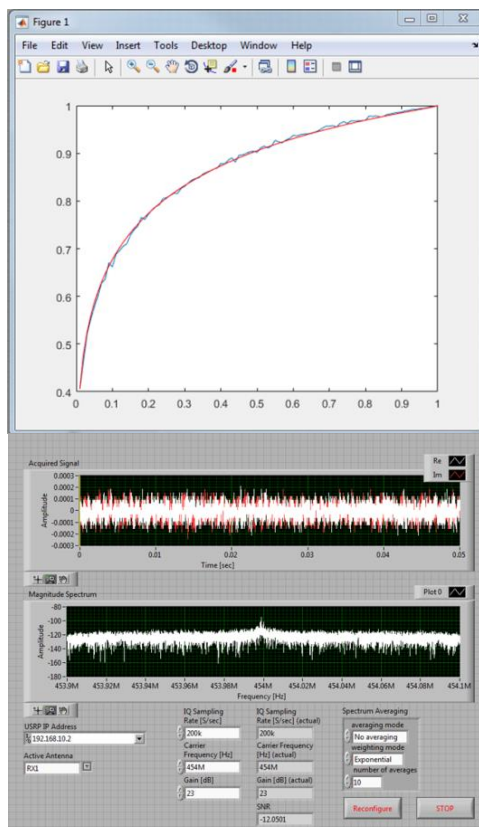


Figure 9. ROC curve between Pd, Pf at SNR = 12.0501 dB

References

- [1] J. Mitola and G. Q. Maguire, "Cognitive radio: Making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, 1999.
- [2] D. Ariananda, M. Lakshmanan, and H. Nikookar, "A survey on spectrum sensing techniques for cognitive radio," *2009 Second International Workshop on Cognitive Radio and Advanced Spectrum Management*, 2009.
- [3] "USRP," - National Instruments. [Online]. Available: <http://www.ni.com/sdr/usrp>. [Accessed: 13-Sep-2016].
- [4] N. Abbas, Y. Nasser, and K. E. Ahmad, "Recent advances on artificial intelligence and learning techniques in cognitive radio networks," *J Wireless Com Network EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, 2015.
- [5] M. Bkassiny, Y. Li, and S. K. Jayaweera, "A Survey on Machine-Learning Techniques in Cognitive Radios," *IEEE Communications Surveys & Tutorials IEEE Commun. Surv. Tutorials*, vol. 15, no. 3, pp. 1136–1159, 2013.
- [6] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*. Cambridge, MA: MIT Press, 1998.
- [7] T. Jiang, D. Grace, and Y. Liu, "Two-stage reinforcement-learning-based cognitive radio with exploration control," *IET Communications*, vol. 5, no. 5, pp. 644–651, 2011.
- [8] T. Jiang, D. Grace, and Y. Liu, "Performance of cognitive radio reinforcement spectrum sharing using different weighting factors," *2008 Third International Conference on Communications and Networking in China*, 2008.
- [9] M. A. K. Abdulsattar and Z. A. Hussein, "Energy Detector with Baseband Sampling for Cognitive Radio: Real-Time Implementation," *Wireless Engineering and Technology WET*, vol. 03, no. 04, pp. 229–239, 2012.
- [10] N. Sharma, D. B. Rawat, B. B. Bista, and S. Shetty, "A Testbed Using USRP(TM) and LabView(R) for Dynamic Spectrum Access in Cognitive Radio Networks," *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*, 2015.
- [11] Y.-C. Liang, Y. Zeng, E. Peh, and A. T. Hoang, "Sensing-Throughput Tradeoff for Cognitive Radio Networks," *2007 IEEE International Conference on Communications*, 2007.
- [12] Y.-C. Liang, Y. Zeng, E. C. Y. Peh, and A. T. Hoang, "Sensing-throughput tradeoff for cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 4, pp. 1326–1337, Apr. 2008.



Pitcha RUNGSAWANG was born in Petchaburi, Thailand on May 30, 1992. I graduated with a B.E. degree in computer engineering from King Mongkut's Institute of Technology Ladkrabang Bangkok, Thailand, in 2014. I'm interested in cognitive radio based on machine learning, signal intelligence, and wireless communication.



Amnach KHAWNE was born in Suphumburi, Thailand, on August 6, 1974. He received the B. E. degree in telecommunication engineering and the M. E. degree in electrical engineering both from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 1998 and 2002, respectively. He received the Ph.D degree in Interdisciplinary Studies from Tokai University, Japan, in 2011. He is currently a lecturer in faculty of engineering, King Mongkut's Institute of Technology Ladkrabang. His research interests include image watermarking, image compression, and network security.