A FLEXIBLE AND EXTENSIBLE COGNITIVE RADIO TEST SYSTEM (CRTS)

Junsung Choi* (Virginia Tech, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, U.S.A.; choijs89@vt.edu); Sayantan Guha* (Virginia Tech, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, U.S.A.; sayantg@vt.edu); Ferdinando M. Romano* (Virginia Tech, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, U.S.A.; fmromano@vt.edu); Tonya L. Smith-Jackson (North Carolina A&T State University, Department of Industrial and Systems Engineering, Greensboro, NC, U.S.A.; tlsmithj@ncat.edu); Carl B. Dietrich (Virginia Tech, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, U.S.A.; cdietric@vt.edu) *Choi, Guha, and Romano share first authorship

ABSTRACT

The multitude of possible approaches to a given cognitive radio application demands an ability to compare the functionality of different cognitive radio systems under a variety of operational conditions. The cognitive radio test system (CRTS) can evaluate the performance of multiple cognitive radios under a range of stressful environments. The CRTS includes communication and link quality feedback between a virtual or SDR-based radio transmitter and receiver, and the ability for the transmitter to switch among multiple predefined operating modes based on the feedback it receives from the radio receiver. Modular implementation using C++ enables efficient implementation of machine learning and optimization techniques, testing of multiple adaptive controllers or cognitive engines (CE), and testing with multiple signal environments that include different combinations of noise and interfering signals. Signal environment characteristics including temporal behavior are parameterized to allow automated generation of multiple operational scenarios of varying complexity to challenge and evaluate CE performance, while hardware modifications such as introduction of filters, preamplifiers, or attenuators enable assessment of the effects of RF front end hardware on cognitive radio performance.

1. INTRODUCTION

As the field of cognitive radio (CR) develops, the need for more sophisticated testing and measurement of CRs is increasing. The multitude of possible approaches to a given CR application demands an ability to compare the functionality of different CR systems under a variety of situations. Such comparisons require a system to evaluate individual CR performance. In addition, testing CEs under

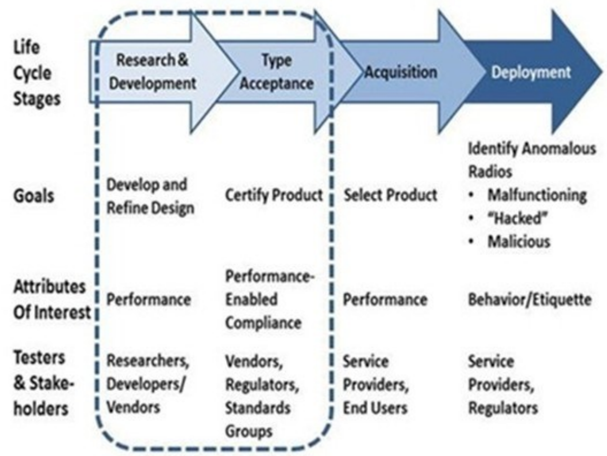

Figure 1. Cognitive radio testing and evaluation requirements throughout the product life cycle

realistic stressful environments is helpful in providing an accurate performance evaluation. A Cognitive Radio Test System (CRTS) is such a system that can evaluate the performance of multiple cognitive radios under a variety of stressful environments.

Like the system described in [1], the CRTS contains a transmitter, a noise source, a receiver, and a feedback path. CRTS includes the ability for the CE to switch the transmitter between predefined operating modes based on the feedback it receives from the radio receiver. CRTS uses a modular design to enable testing of various CEs under a variety of noise and interference scenarios, where each CE and each scenario is specified by a configuration file as described in Section 3.

From a systems engineering viewpoint, commercial CRs can be considered as industrial products that require efficient testing at all stages of the product life cycle [9]. These stages include research and development, type acceptance for regulatory and standards compliance, product selection and acquisition, and deployment and subsequent

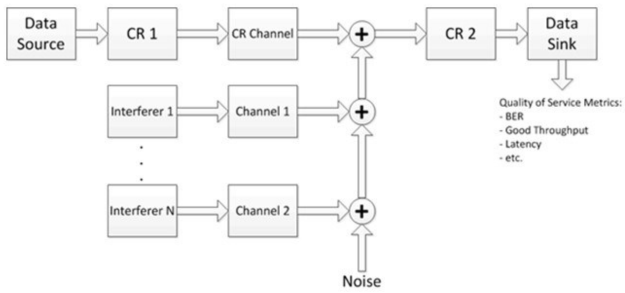

Figure 2. Example configuration for CR test infrastructure (reverse channel from CR2 to CR1 omitted for clarity)

operation. This integration of the product within the socio-technical system leads to more robust and effective solutions. CR testing goals, CR and CR network (CRN) attributes of interest, and testers and stakeholders involved are shown in Figure 2 for each stage of the product life cycle. In this project we are initially concerned with testing and evaluation in the two phases of the product life cycle that are highlighted in Figure 1: (1) testing in support of research and development, which helps researchers and developers develop and refine CR designs to improve performance; and (2) type-acceptance testing used by vendors, regulators, and standards groups to verify compliance of CRs with regulatory and voluntary standards

To provide a realistic assessment of cognitive radio performance, a system needs the capability to test a cognitive radio under more than one signal environment or noise and interference scenario. Moreover, for efficiency, evaluating performance of several CEs in rapid succession is necessary. Through its ability to test many CEs in a variety of selected noise and interference scenarios, CRTS can evaluate CR performance efficiently under more realistic environments, and is an effective test bed system for cognitive radios.

The dynamic interplay of changing spectrum environments, performance goals, and reconfigurable capabilities implies that creating a generic benchmarking method for CRs is not trivial. In fact, CR algorithms such as those used for dynamic spectrum access (DSA) provide CRs with artificial intelligence. CR performance can be improved over time if a learning capability is implemented in CRs with a reliable standardized measurement for ongoing evaluation purposes.

CRs have capabilities that are inspired by human cognitive capabilities, making it possible to apply psychometric methods to CR performance assessment [2, 3, 4, 6]. Psychometric methodologies such as item-response modeling (IRM) and cognitive diagnostic modeling (CDM), enable measurement of human/radio cognition, e.g., by analyzing humans' (or radios') observable response to a series of test items [2, 3, 4, 6, 12]. Based on observations and behaviors, radios can be evaluated or classified into appropriate groups or classes. In addition to psychometric methods, a wide range of other statistical modeling approaches may also be applied to cognitive radio evaluation, including use of Bayesian classifiers [5]. Regardless of the methodology used for modeling and analysis of CR performance, a test bed is needed that enables controlled CR simulations and measurements.

In addition to the need to assess CR performance in the research and development stage, there is a strong need for a testing methodology for assessing CR performance and performance-enabled regulatory and voluntary standards compliance of CRs and by extension, of CRNs

2. HARDWARE

This section describes in detail the hardware architecture of the CRTS. CRTS has been written in C++, and includes software-simulated interfaces with a transmitter, a noise/interference source, and a receiver, as well as a feedback loop. It also provides the ability to change the parameters of scenarios and CEs through configuration files.

The hardware interface of CRTS is in progress and has similar aspects to other test beds, containing a transmitter, a receiver, noise/interference source, and feedback loop.

The CRTS hardware will consist of three main sections: a transmitter, a noise source, and a receiver. The first two are controlled by a single computer that contains the configuration files of the CE and the scenario and is connected to its own USRP. The receiver is similarly controlled by a PC attached to a USRP but also sends feedback to the CE over a TCP/IP connection. Figure 3 shows a basic schematic of CRTS.

2.1. Transmitter (Tx)

The transmitting computer of the CRTS simulates sending data to the receiver by manipulating baseband samples. Once the hardware is fully integrated, data is sent to the

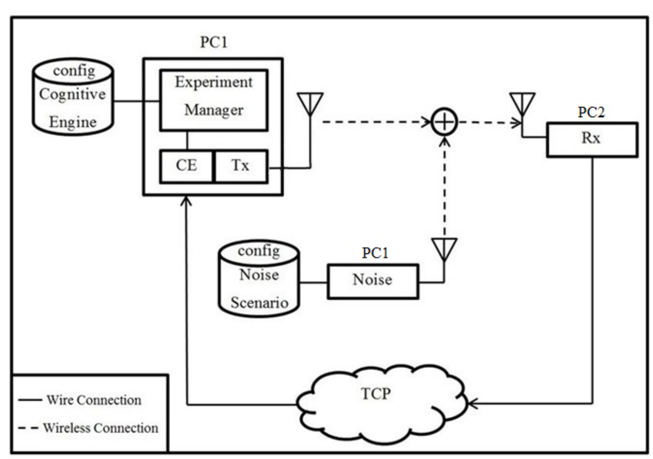

Figure 3. Diagram of CRTS

receiver via the transmitting and receiving USRP’s, connected by SMA cable or, eventually, wirelessly. The receiving USRP is in turn controlled by a separate computer.

The transmitting computer contains CEs which decide parameters such as modulation scheme, data rate, and center frequency of the transmission. Decisions are made based on feedback from the receiver as well as from the user-defined configuration settings. The Transmitter PC, which also controls the noise/interference, acts as the top module for the test system itself. It selects what kinds of CEs and noise scenarios to run, analyzes performance under different noise environments, and finally prints the result.

2.2. Noise and Interference Scenarios

The noise and interference scenarios, controlled by the main PC, send simulated noise and interference over the transmission channel to be added to the transmitter signal. These scenarios currently include Additive White Gaussian Noise (AWGN), and Rician fading, and can be extended to include various types of interference (e.g., single or multiple, narrowband or wideband, unintended or malicious)., all similar to the scenarios used in [1]. Multiple scenarios can be combined in one, and the parameters of each are specified by the configuration files.

2.3. Receiver (Rx), Feedback loop

The receiver is simulated on the main computer. For each frame received, the receiver sends feedback to the CE via a TCP/IP network. The feedback can contain information about the received signal such as power, bit error rate, etc. With the USRP’s integrated into the system, a separate computer controls the receiving USRP and sends feedback over the same TCP/IP network as in simulation mode.

3. SOFTWARE

CRTS is written in C++. Apart from the standard C++ libraries, use has also been made of the liquid-dsp library, for performing digital signal processing operations on data, and libconfig, to allow the program to read from configuration files.

CRTS is distinctive in its use of configuration files to enable definition of interchangeable operational scenarios and cognitive engines. Figure 4 describes the method of operation of the Test System and how configuration files are used.

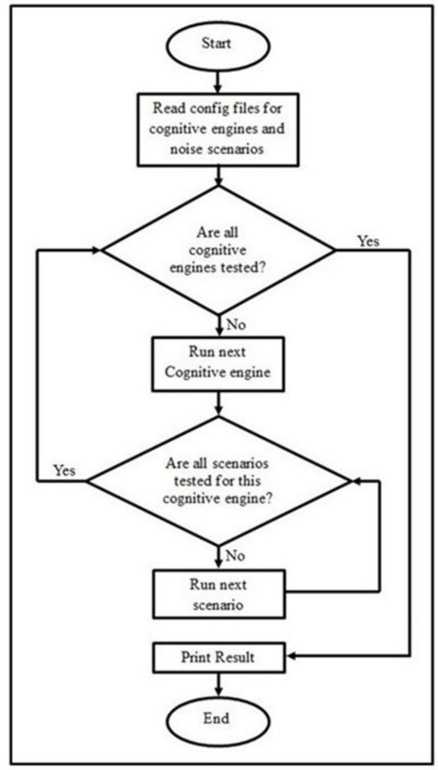

Figure 4. Flowchart of CRTS Process

3.1 Configuration Files

In order to facilitate the testing of multiple cognitive engines under multiple scenarios, two kinds of configuration files are used. The first set contains the characteristics of each cognitive engine, every file describing a separate engine. In each file, the user can specify the cognitive engine parameters such as Default Transmit Power, modulation scheme, or a predefined method of adaptation with its own adjustable parameters. A simple example of such an adaptation is given in Section 4. The second set of files describes the scenarios to be tested. The user can incorporate a single predefined scenario such as Gaussian noise, or combine multiple scenarios into a single new scenario, such as Gaussian noise and fading. Each configuration file allows specification of a scenario’s parameters, such as noise power. Finally there is a pair of master files which specify which cognitive engines to run, and under which scenarios to run them.

The configuration files provide an interface for the user to easily create cognitive engines and the test scenarios and they allow multiple cognitive engines to be tested one after the other, each through multiple scenarios, facilitating analysis and comparison.

3.2. Data collection

While each CE runs, the control computer continuously records data about the CE engine operation, such as the number of frames detected by the receiver, the elapsed time of operation, or the packet error rate. This data can then be used to compare and evaluate the performance of cognitive engines with respect to a variety of metrics.

```
//config1.txt
// Basic Information:
filename = "Master Configuation File for Congnitive Engines";

// Parameters
params =
{
NumberofCogEngines = 1;
cogengine_1 = "config_cog_engine.txt";
};
```

```
//config1.txt
// Basic Information:
filename = "Master Configuation File for Scenarios";

// Parameters
params =
{
NumberofScenarios =2;
scenario_1 = "AWGN4.txt";
scenario_2 = "AWGN10.txt";
};
```

```
//AWGN1.txt
//Basic Information:
filename = "AWGN1";

//Parameters for White Gaussian Noise
params =
{
noiseSNR = 20.0;
};
```

Figure 5. Examples of Configuration File

4. RESULTS

The CRTS was tested with two cognitive engines, each with a goal of successfully sending 1000 error-free packets. Both cognitive engines were tested under four different scenarios: simulated AWGN channels with theoretical Signal-to-Noise ratios of 30, 20, 10 and 5 dB. In each case the cognitive engines started with a default modulation scheme of 128-PSK and, if a specified condition was not met, divided the M-value of the PSK scheme by two; unless it reached M=2 (BPSK) whereafter it would remain unchanged. The first cognitive engine required that the previously sent packet be received without error. The second cognitive engine required that the total packet-error-rate be below 50%. Tables 1 and 2 describe the performance of these two cognitive engines using data from the CRTS and provide simple examples of metrics measurable by the system.

Table 1: Cognitive Engine 1

Channel SNR (dB)	30	20	10	5
Final Modulation Scheme	16PSK	BPSK	BPSK	BPSK
Time elapsed (seconds)	0.9767	1.6236	1.6624	33.2141
Final Packet-Error-Rate	0.1996 %	0.3984%	0.4975%	94.47%
Number of Packets Detected by Receiver	1002	1004	1005	18081

Table 2: Cognitive Engine 2

Channel SNR (dB)	30	20	10	5
Final Modulation Scheme	16PSK	8PSK	BPSK	BPSK
Time elapsed (seconds)	1.0206	1.2004	1.9296	31.7150
Final Packet-Error-Rate	0.2991 %	0.3984%	0.5964%	94.50%
Number of Packets Detected by Receiver	1003	1004	1006	18188

5. CONCLUSIONS AND FUTURE WORK

CRTS will facilitate standardization of cognitive radio metrics and investigation of methodologies for efficient test and evaluation of cognitive radios and, eventually, of cognitive radio networks. Further, there are numerous other possible applications in which the CRTS system can be used; for example, in a Cognitive Engine performance contest, where it can provide a uniform test system that will analyze and compare the performance of different CEs under noise, interference, fading and other effects.

Although the CRTS provides a wide range of features for testing cognitive engines, various improvements to the system structure can be made. For instance, the current implementation of the CRTS only has unidirectional over-the-air transmission. This can be extended to a duplex connection where two separate nodes act as transceivers and each have CE capabilities. This would allow the sending of feedback over-the-air, instead of over a separate TCP/IP network, and each node can make decisions based on received signals at each side.

Another possible improvement is to implement an extended version of the CRTS in large-scale SDR/CR test beds such as CORNET, ORBIT, etc., which would allow investigation of approaches for efficient testing of a whole network of cognitive radios.

However, even before these improvements can be incorporated, CRTS already has a demonstrated ability to compare a variety of cognitive engines under various circumstances, thus enabling further research into development and evaluation of high performance cognitive radios.

6. REFERENCES

[1] A. E. Amanna, "Statistical experimental design framework for cognitive radio," Dissertation submitted to Virginia Polytechnic Institute and State University, March 2012

[2] C.B. Dietrich, E.W. Wolfe, and G. Vanhoy, "Cognitive radio testing using psychometric approaches: applicability and proof of concept study," Analog Integrated Circuits and Signals Processing, Vol 73, No. 2, pp. 627-636, 2012.

[3] C.B. Dietrich, E.W. Wolfe, and G. Vanhoy, "Cognitive Radio Testing using Psychometric Approaches,"SDR11/WinComm, 2011.

[4] C.B. Dietrich, E.W. Wolfe, and G. Vanhoy, "Evaluation of multi-objective optimizers for cognitive radio using psychometric methods: Analysis using unidimensional and multidimensional Rasch models," 7th International ICST Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM), pp. 326-331, 2012.

[5] D. DePoy and T. Bose, "Predictive behavior classification for cognitive radio: introduction and preliminary results," ICST CROWNCOM, June 2012.

[6] E.W. Wolfe, C.B. Dietrich, and G. Vanhoy, "Application of the Rasch model to measuring the performance of cognitive radios," Fifth International Conference on Probabilistic Models for Measurement, 2012.

[7] J. Insa-Cabrera and J. Hernandez-Orallo, "Comparing humans and AI agents," Conference on Artificial General Intelligence (AGI 2011), Mountain View, CA, 2011.

[8] J. Mitola and G.Q. Maguire, "Cognitive radio: making software radios more personal," IEEE Personal Communications, vol 6, no. 4, pp. 13-18, August 1999.

[9] M.T. Reynolds, *Test and evaluation of complex systems*, Wiley, Nov 1997.

[10] S. Bringsjord and B. Schimanski, "What is artificial intelligence? Psychometric AI as an answer," IJCAI 2003 Conference, 2003.

[11] S. Haykin, "Cognitive radio: brain-empowered wireless communications," IEEE Journal on Selected Areas in Communications, vol. 23, no. 2, pp. 201-220, February 2005.

[12] Y. Zhao, S. Mao, J.O. Neel, and J.H. Reed, "Performance Evaluation of Cognitive Radios: Metrics, Utility Functions, and Methodology," Proceedings of the IEEE, Vol. 97, No. 4, pp. 642-659, April 2009.