COGNITIVE GATEWAY DESIGN TO PROMOTE UNIVERSAL INTEROPERABILITY

Qinqin Chen, Charles W. Bostian
(Center for Wireless Telecommunications, Wireless @ Virginia Tech, Blacksburg, VA 24061; {chenq, bostian}@vt.edu)

ABSTRACT

Cognitive gateways (CG) are conceived as a kind of special cognitive radio (CR) node that interconnects different systems. This paper presents a CG design which will facilitate universal interoperability between incompatible waveforms used by a variety of heterogeneous communication systems. In specific scenarios, a CG can act as a signal repeater, a waveform gateway or a network gateway to provide an extended service coverage area. The term "waveform" is defined as a protocol stack specification suite, namely a set of parameters describing the format of a communication signal (PHY) and its related processing protocols (MAC, LLC, NET, etc.).

We detail the primary steps executed in a complete CG and address the key technologies used in each step. A simplified prototype has been designed and tested on the platforms available in our lab to prove the functionality of the proposed CG.

1. INTRODUCTION

Hardware based transceivers designed for different waveforms cannot interoperate with each other. This is a particular problem for public safety and military communications systems.

Generally, there are two solutions to the interoperability problem: requiring that (1) all the communication systems comply with a common standard, or (2) each communication node is able to accommodate all the existing waveforms. A good example that combines the ideas in (1) and (2) is the Project 25 (P25) system. "Project 25 is a public safety communications standard dedicated to ensuring interoperability in communications."[1] P25 introduces specific definitions for critical system interfaces, which include the Common Air Interface (CAI), the Inter-RF Subsystem Interface (ISSI), the interface for the world-wide PSTN, the interface for host and network (such as TCP/IP) connectivity. The full implementation of P25 depends on the ubiquitous usage of P25-compliant radio systems. This may take many years to be accomplished because the ISSI and other interfaces are still under development, and currently manufacturers mainly provide radios supporting only the P25 CAI standards.

Cognitive radio (CR) is an attractive technology that belongs to the second solution. It is conceived as a flexible, reconfigurable radio that is guided by intelligent processing to sense its surroundings, learn from experience and knowledge, and adapt the communications system to improve the use of radio resources and provide desired quality of service. Our CWT (Center for Wireless Telecommunications at Virginia Tech) team builds cognitive radios as a software defined radio (SDR) operating under the control of an intelligent software package called a cognitive engine (CE) [2][3][4].

A CR is capable of generating any waveforms supported by the available radio hardware and software resources. Ideally, it can accommodate all the existing waveforms. So, if the technologies have been advanced enough that all the existing communication systems could be replaced by CRs, seamless communications at anytime, anywhere will become reality. However, current technologies fall behind the conception for cognitive functionalities, so different communication systems will co-exist for a long time even if they cannot interoperate with each other. Therefore, we propose a cognitive gateway (CG) to bridge incompatible waveforms; it is designed on the basis of CR concept.

A gateway in a communications network is a network node equipped for interfacing with another network that uses different protocols. Gateways operating at any layer of the OSI model can also be called protocol converters. As a more intelligent node able to interconnect different systems, a CG needs to accomplish much more than just converting protocols. A complete CG system is composed primarily of eight modules as shown in Fig. 1: These are the waveform identifier, scenario analyzer, waveform and user databases, decision maker, central controller (including logic link controller), generic system API, and waveform converter. This architecture also follows CWT's cognition loop [2]-[5] and is quite similar to that of the CRs developed in our laboratory. The biggest differences lie in four aspects. (1) Using its available hardware and software resources, a CG is responsible for establishing as many communication links with a specified quality of service as needed between incompatible waveforms, so (2) it needs to identify the types of both source and destination waveforms, and (3) it requires a protocol to manage the establishment, maintenance, and termination of logic links. (4) It needs a procedure for user registration and authentication. Thus, the design for CGs has some difference from the discussion presented in [2]-[4], although similarities inevitably exist. The key technologies used in CG will be addressed in detail.

The remainder of this paper is organized as follows: Section 2 begins with an introduction to the applications of CG, followed by a system architecture overview for CG. We address the implementation details for the critical modules of a complete CG system in Section 3. Section 4 describes a prototype developed in CWT to validate the functionalities of the proposed CG. Our conclusions are made in Section 5.

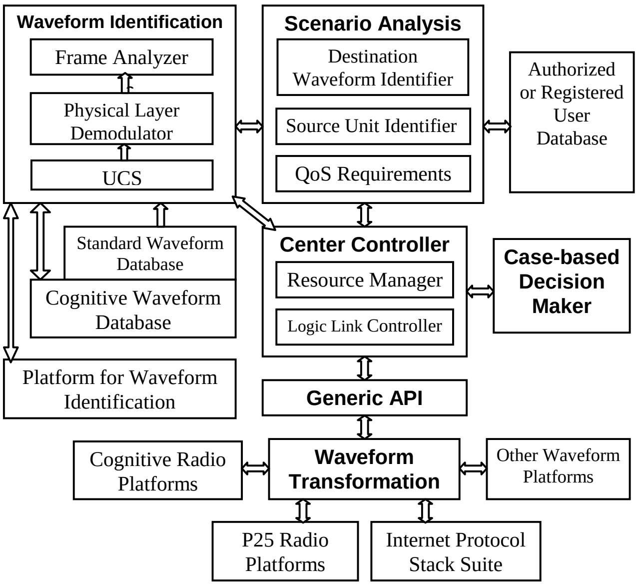

Figure 1. Cognitive Gateway Block Diagram

2. SYSTEM OVERVIEW

In this paper, we locate CGs in a network incorporating heterogeneous communication nodes. Those that comply with the same or similar standard are able to communicate with each other directly, via a repeater, or an intra-subnet gateway; while those following different standards need an inter-subnet gateway to bridge them. According to our earlier "waveform" definition, repeater, intra-subnet gateway, and inter-subnet gateway all implement the same function — waveform transformation. These are specific forms of CG in different application scenarios and help extend the communications coverage from intra-networks to inter-networks. In Fig. 2, we describe a scenario where the CG acts as an inter-subnet gateway to bridge four kinds of systems, including three standard systems, i.e., conventional public safety systems, P25 enabled systems, IP-based wireless LANs, and user-developed CR systems.

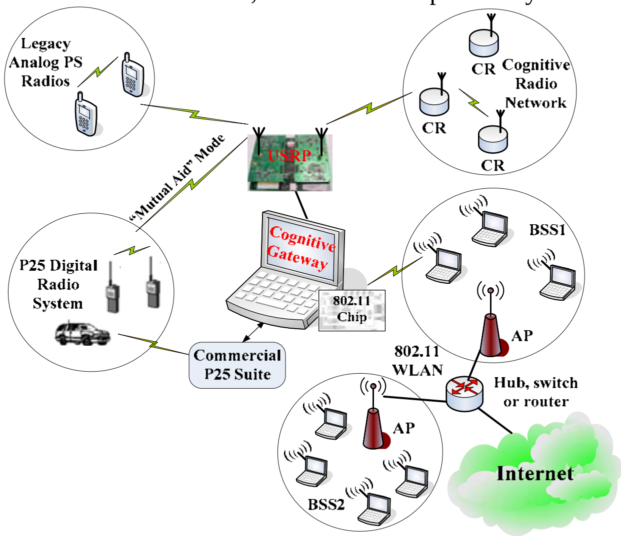

Figure 2. A Scenario Using A Cognitive Gateway

A gateway is the intersection point for different systems. It has multiple identities and interfaces. The CG shown in Fig. 2 is wireless. It uses software/hardware (such as GNU Radio [6] plus Universal Software Radio Peripheral (USRP) [7]) similar to wireless network cards to transmit and receive signals over the air. Its interface to an IP-based network should have an IP address; its interface to a P25 system should have an address and identification which can be recognized by P25 radios; its interface to a CR network should be known to CR nodes. For each subnet using this CG, if a node within one subnet wants to send data to another node belonging to a different subnet, it only needs to send the data to the CG which can reach the desired destination. This principle is analogous to that used in IP-based network: a very popular example is connecting a LAN to the Internet or a WAN [8]. But a CG has to do much more than replacing the original source address with the new one. A CG needs to recognize the incoming waveform to determine whether it is a signal it should forward or drop, and which waveform platforms it should launch to implement waveform transformation. The major tasks of a CG includes: waveform identification, scenario analysis, decision making, link control, and waveform transformation.

A CG block diagram is given in Fig. 1. Basically, the waveform identification module consists of signal classifier, physical layer demodulator, and frame analyzer. It works with the two waveform databases (a standard waveform database and a cognitive waveform database) to identify the waveform coming into the CG. The signal classifier determines the physical layer parameters like carrier frequency, bandwidth, modulation type, and symbol rate. In most cases, these parametric values are sufficient for waveform identification. If not, the physical layer demodulator will be configured by these parametric values to extract the link layer frames; then, the frame analyzer will deduce the frame format to help decide the source waveform type. Meanwhile, the source and destination addresses or identifiers will be extracted and fed into the next block—the scenario analyzer. Referring to the user database, the scenario analyzer identifies the two ends of a link that the CG needs to interconnect. We call this link establishment request an *application*. Applications from authorized users will be placed in the waiting queue of the central controller. After the scenario analysis step, a resource manager and a link controller provide necessary information for the decision maker. Based on the decision result, the central controller will allocate appropriate resources to implement the applications and meet their priorities and QoS requirements to the best of their ability. Then, the system configuration profiles and necessary control commands will be generated to launch corresponding platforms or components for implementing waveform transformation, thereby establishing communication links between different waveform platforms.

Because of its wireless nature, a CG can only provide service to the users within its effective range. Its user database contains all the users within its service group. This service group mainly includes pre-authorized public safety radios, wireless IP nodes, and registered CR nodes. When a new CR node appears, it can receive a beacon message broadcast periodically by the CG. By sending request messages to the CG, it is able to join and quit the service group, and the user database will be updated accordingly.

Our goal is to make the whole system automatic after the initial human setup. However, user manipulations are still required when a CG is used with the conventional public safety system because the traditional analog radio signal does not carry destination information and we need to adapt the existing public safety analog systems rather than change them in accordance with our requirements.

A CG can be implemented in a distributed manner, where the modules connected to the central controller may be located in the same or different hosts. As shown in Fig. 1, any two blocks or modules that connect with each other need to exchange information or convey information from one to another. “Sockets” can serve as the tunnel for the information transmission. As the scheduler of the whole CG, the central controller uses a general method to describe the attached modules. This description may include a set like (module name, host name or address, port number). A distributed implementation enables the concurrent operation of multiple functional modules.

3. SYSTEM IMPLEMENTATION

Before diving into the details for each module, we first show how to present a waveform because waveforms are the initiators through the whole procedure of a CG.

3.1. Waveform Representation

The waveforms used in standard systems have been well defined, but a CR system is distinguished by waveform agility and there is not a uniform specification for its mechanisms or protocols. Taking into account that an IP-based network is the backbone infrastructure and that the five-layer OSI model has been widely and successfully used, we decided to describe the waveforms according to the five-layer protocol stack architecture. Our definition for “waveform” in this paper is based just on the above considerations.

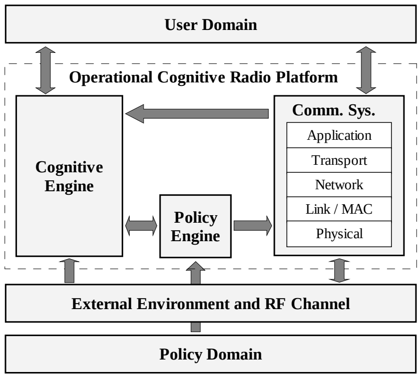

Figure 3. Generic Cognitive Radio Architecture (from [3])

In our discussion, we assume that the CR nodes constituting a CR network follow the generic CR architecture shown in Fig. 3. This architecture determines the waveform format for CR. Actually it is challenging to develop (1) CR nodes with complete stack architecture and functionalities, and (2) necessary mechanisms/protocols for communications among nodes following the same or different standards. These two tasks are very important to CG design, but they concern too many issues, which cannot be clearly addressed in a few words. Hence, we only list here the key issues to be considered as follows.

- Supported applications and corresponding QoS, security and reliability
- Transport layer protocols
- Routing algorithms
- Addressing and mobility management
- Unit identification, Database management
- Channel utilization scheme, logic link control
- Frame architecture and message formats for inter- and intra-system communication
- REM (radio environmental map) acquisition

In Table 1, we give a reference format that can represent the typical existing waveforms like standard

802.11b and 802.11g, and a P25 waveform complying with the standard CAI. The five layers are denoted as PHY, LINK, NET, TRAN, APPL, respectively. The given format has a hierarchical architecture, where the concrete format of the lower tier is subject to change based on the higher tier specification. For example, the traditional analog public safety waveform does not have NET and TRAN layers at all. This architecture possesses several advantages: (1) clear hierarchy/layering is efficient for parameter extraction at the system configuration stage; (2) it is flexible to adapt different waveforms and open for future modification. Actually, many parameters of the standard waveforms use fixed default values, and some “knobs” outlined in Table 1 do not exist in the standard waveforms. Therefore, when a waveform profile containing these “knobs” is generated for system (re)configuration, a parser does not need to extract all the parametric values. This will speed up the system configuration step.

Table 1. Reference Waveform Format (a sample)

Tiers (high→ low)				Parameters (or Knobs)
Waveform Type Options: CR, WiFi, P25, Conventional Public Safety	TX	PHY	RF	Tx carrier frequency, Tx power
			MOD	Modulation type and index, Symbol rate, Roll off, Differential coding
			FEC* (e.g. RS)	Number of input symbols, Number of output symbols
		LINK	MAC* protocol (e.g. CSMA)	Carrier sense threshold, Contention window, Minimum back-off delay
			Frame	Frame size, Frame type
		NET	Protocol	Network protocol (IPv4 or IPv6) IP address allocation protocol Routing protocol
			IP packet	Packet size
		TRAN	Protocol	TCP or UDP
		APPL	Service* type	Application protocol (e.g. HTTP, SMTP)
			encryption	Encryption key
	RX	Similar as the format for TX		

* Type options for Medium Access Control (MAC) Protocol: CSMA/CA, PTT, ALOHA etc. Type options for FEC: convolutional coding, block coding (including Reed-Solomon (shortened as “RS” in Table 1, BCH, Golay, Hamming). Service type could be audio, video, multimedia, data etc.

In addition, this waveform representation method is platform independent. No matter which vendor the device is from, no matter whether the CR is built on GNU Radio, IRIS (Implementing Radio in Software) [9], or OSSIE (Open Source SCA Implementation - Embedded) [10], this format works. In CWT, we select extensible markup language (XML) to describe a waveform.

3.2. Waveform Identification and Scenario Analysis

The waveform identification module not only identifies the incoming waveform type but also provides the REM to the central controller as the reference for medium access control. The output of waveform identifier will be a subset of the complete waveform representation addressed in Section 3.1; while the information fed into the central controller by the scenario analyzer may be a set like (node A: waveform A: network A, node B: network B, high priority), which means node A from network A, which is using waveform A, requested to establish communication link with node B in network B, and this application has a high priority. Note that waveform A is a combined entry from the waveform database. If a signal from an unwanted user is detected, the CG will discard it.

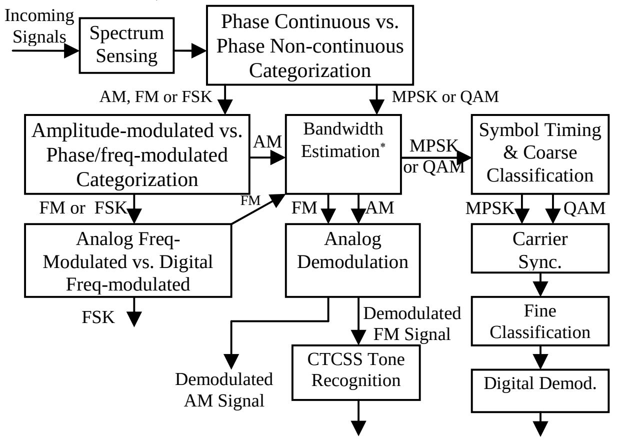

Figure 4. UCSD System – Function Block Diagram

(*Bandwidth estimation varies for different modulation groups.)

The waveform identification step employs our newly developed universal classification, synchronization and demodulation (UCSD) system [11] to implement the physical-layer signal recognition, and the physical layer demodulation when necessary. Our UCSD system currently supports a variety of modulations, including analog AM and FM, and digital PSK, QAM, FSK. The system block diagram is shown in Fig. 4. It can automatically interpret features of the received signal to accomplish classification, synchronization and demodulation without knowing any prior modulation information. As we add more modems to our software repository, OQPSK and OFDM signals will also be included [12].

Under the traditional static spectrum policies a spectrum sensing module, which is able to provide signal location information in the frequency domain (including center frequency and bandwidth) is almost sufficient for waveform identification. However, the introduction of market based spectrum policies and the adoption of CR, dynamic spectrum access (DSA) result in the occurrence of new waveforms and increase the difficulty and complexity for identifying standard waveforms. Thereby, in some cases, information in addition to the physical-layer parameters must be extracted from the link layer frames to provide more accurate waveform identification, and also to guide the posterior steps. For example, a CG needs to know the destination address and waveform in order to determine how the API should be configured, and which interface the received data should be forwarded to. We therefore take advantage of a multi-layer waveform identifier.

The platform for waveform identification can be a stand-alone MATLAB-enabled Anritsu MS2781A Signature Signal Analyzer [13] running Windows, a sensor implemented over GNU Radio plus USRP in Linux and embedded in the same host as the central controller, or a Lyrtech SFF SDR platform [14] connected to a PC.

3.3. Database

We have four databases serving for the CG: standard waveform database, CR waveform database, user database, and system resource database contained in resource manager. The waveform databases are organized as hierarchical architectures with "knobs" whose values are obvious for waveform differentiation. Instead of outlining all the combinations, we list all the possible values for each "knob". The user database includes IDs for authorized users and the address sets used in different subnets. Since a CG provides services for wireless mobile users within its contributing range, the users' mobility will result in the update of this database. The system resource database contains the usage situation of CPU, memory, power, and waveform platforms. Thus, it is a dynamic database.

3.4. Generic API and Waveform Transformation

The major task of the generic API in Fig. 1 is to convey information between the central controller and the waveform converter. A reconfigurable, flexible API that balances among generality, efficiency, and complexity is an ideal choice for CG. Generally, the information for hardware configuration, waveform pair specification, and system link behavior control is transmitted via this generic API. Just like the idea for a generic waveform representation in Section 3.1, this generic API targets all the layers of a protocol stack and thus can be configured to meet the specific requirements for different applications.

Waveform transformation could be much easier if we have corresponding waveform modems. That means we can first demodulate the received signal and extract the payload, then encapsulate and modulate the payload into another waveform format, then transmit it. Besides the payload, we also need to extract the necessary information required for re-transmission (e.g. destination address, destination ID). Repeaters and intra-subnet gateways usually bridge the waveforms which have different physical layer parameters, such as carrier frequency, modulation, symbol rate etc. As to the intra-subnet gateway exploited in Fig. 2, instead of making great efforts to develop the modems for standard P25 waveforms and 802.11b, g waveforms under the open source software environment like GNU Radio, we directly make use of inexpensive WiFi chips for PHY- and LINK-layer processing, and the built-in protocol stack, well supported by operating system, for upper-layer processing. We also use the E.F.Johnson 5300 ES Series Mobile Radio platform [15], which provides an interface to be controlled by PC. If GNU Radio is assumed as the software platform for CR implementation, we can build a TUN/TAP (a virtual point-to-point and Ethernet device) [16] to create an interface "gr0", which is similar to the interface "eth0" for WiFi. The information for hardware configuration is generated based on current application (i.e. the waveform pair) and link status.

A complete generic API can be simply expressed as: (waveform A: platform A, waveform B: platform B, link control command). The waveform representation refers to Table 1, while the waveform platform format refers to [17]. Both the waveform profile and its corresponding platform profile will be greatly simplified if the waveform pair uses standard waveforms. A block diagram shown in Fig. 5 describes the link status control and the generic API.

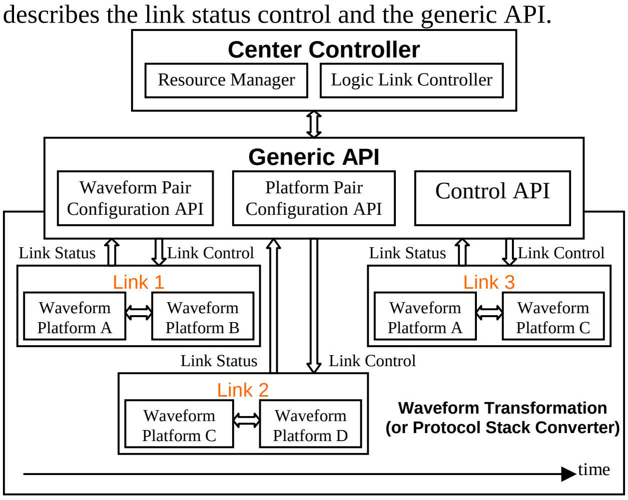

Figure 5. Generic API and Link Control

3.5. Link Control

One of the big challenges for CG implementation is the link control. A CG may manage multiple links. A finite state machine (FSM) based link control protocol is essential to smooth link configuring, switching, establishment, maintenance, and termination. We need to determine the states and actions for FSM. The central controller sends a command (such as load, reload, start, stop, pause) to guide the target waveform platforms' behavior while the waveform converter returns link status to the central controller. The FSM design also takes into account the

MAC protocols used for each waveform transformation link. The format of link status can be expressed as: Link 1 (waveform platform A→ waveform platform B): busy. The number of links that a CG can handle during a given period of time depends on its capabilities and available software/hardware resource. In some cases, the sensor of a CG competes for the waveform platform for CR. In addition, a CG for a wireless communication system is different from the gateway used in wired networking system in the aspect that the multiple waveform platforms cannot be simultaneously employed if the signals are transmitted at the same or close channels due to the self-interference problem. For those links that can coexist, multiple threads are created. The performance of a CG will be evaluated by the metrics *link throughput*, *delays*, and *packet loss rate*.

4. PROTOTYPE

A prototype shown in Fig. 6 will be built to verify the functionalities of proposed CG. A CG node is implemented on the basis of PSCR (public safety cognitive radio) [17], "Smart" Radio [18], and UCSD developed in CWT.

There are two types of cognitive radios in Fig. 6. Type 1 CR has the functions as follows: (1) able to classify FM, BPSK, QPSK, 8PSK signals and adapt to these modulations; (2) primary user detection and waveform switching; (3) routing enabled and identification-embedded; (4) carrier sensing before transmission. Type 2 CR is a CG able to recognize and bridge the waveforms from traditional analog public safety radios, p25 radios, type 1 CR, and WiFi nodes.

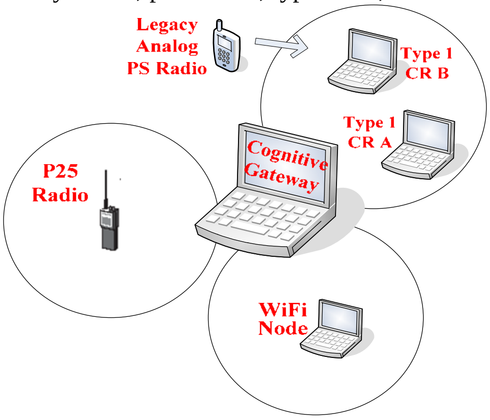

Figure 6. Demonstration Prototype for CG

5. CONCLUSION AND FUTURE WORK

In this paper, we propose a cognitive gateway to promote universal interoperability. Specifically, a hierarchical waveform representation format is described after considering the trade-offs among generality, flexibility, and efficiency. The design for waveform identification, database, generic API, and link control are detailed. Our future work will focus on: (1) quantitative performance evaluation for CG; (2) investigation on appropriate platforms for CG.

6. ACKNOWLEDGEMENT

This work is supported by Award No. 2005-IJ-CX-K017 awarded by the National Institute of Justice (NIJ), Office of Justice Programs, US Department of Justice and also supported by the National Science Foundation (NSF) under Grants CNS-0519959. The opinions, findings, and conclusions expressed in this publication are those of the authors and do not necessarily reflect the views of NIJ and NSF. The authors would like to thank our CWT colleagues for their contribution to this project.

7. REFERENCES

[1] Project 25, http://www.p25.com/
[2] B. Le, "Building a Cognitive Radio: From Architecture Definition to Prototype Implementation," *Ph.D. dissertation*, Virginia Tech, 2007.
[3] T. W. Rondeau, "Application of Artificial Intelligence to Wireless Communications," *Ph.D. dissertation*, VT, 2007.
[4] B. Le, T. W. Rondeau, and C. W. Bostian, "Cognitive radio realities," *Wireless Communications and Mobile Computing*, vol. 7, no. 9, pp. 1037-48, November 2007.
[5] J. Mitola III, "Cognitive radios: An integrated agent architecture for software defined radio," *Ph.D. dissertation*, Royal Institute of Technology, 2000.
[6] GNU Radio, http://gnuradio.org/trac
[7] USRP, http://www.ettus.com/
[8] J. F. Kurose, K. W. Ross, "Computer Networking: A Top-down Approach Featuring the Internet", the 3rd Edition, Pearson Education, Inc. 2005.
[9] P. Sutton, L. E. Doyle and K. E. Nolan, "A Reconfigurable Platform for Cognitive Networks," in *Proceedings of the 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, June 8-10, 2006.
[10] OSSIE, http://ossie.mprg.org/
[11] Y. Wang, Q. Chen, and C. W. Bostian, "Universal Synchronizer Design for Cognitive Radio," *in SDR Forum Technical Conference*, Denver, CO, 2007.
[12] Y. Wang et al, "OFDM Signal Classification and Synchronization for Cognitive Radio Systems," accepted for *SDR Forum Technical Conference*, 2008.
[13] Anritsu MS2781A Signature Signal Analyzer, http://www.us.anritsu.com/products
[14] Lyrtech Small Form Factor (SFF) SDR platform, http://www.lyrtech.com/
[15] EFJohnson 5300 ES Mobile Radio, http://www.efjohnson.com/products/5300ES.asp
[16] TUN/TAP, http://vtun.sourceforge.net/tun/
[17] B. Le, P. Garcia, Q. Chen , B. Li , F. Ge, M. ElNainay , T. W. Rondeau , and C. W. Bostian, "A Public Safety Cognitive Radio Node System," *Software Defined Radio Technical Conference*, Denver, Colorado, November, 2007
[18] M. D. Silvius, F. Ge, A. Yong, A. B. MacKenzie, C. W. Bostian, "Smart Radio: Spectrum Access for First Responders", *SPIE Defense and Security Conference*, Orlando, FL, March, 2008.