ABSTRACT

Title: Spectrum Sensing Security in Cognitive Radio Networks

Awais Khawar, Master of Science, 2010

Supervisor: Dr. Charles Clancy
Department of Electrical and Computer Engineering

This thesis explores the use of unsupervised machine learning for spectrum sensing in cognitive radio (CR) networks from a security perspective. CR is an enabling technology for dynamic spectrum access (DSA) because of a CR's ability to reconfigure itself in a smart way. CR can adapt and use unoccupied spectrum with the help of spectrum sensing and DSA. DSA is an efficient way to dynamically allocate white spaces (unutilized spectrum) to other CR users in order to tackle the spectrum scarcity problem and improve spectral efficiency. So far various techniques have been developed to efficiently detect and classify signals in a DSA environment. Neural network techniques, especially those using unsupervised learning have some key advantages over other methods mainly because of the fact that minimal pre-configuration is required to sense the spectrum. However, recent results have shown some possible security vulnerabilities, which can be exploited by adversarial users to gain unrestricted access to spectrum by fooling signal classifiers. It is very important to address these new classes of security threats and challenges in order to make CR a long-term commercially viable concept.

This thesis identifies some key security vulnerabilities when unsupervised machine learning is used for spectrum sensing and also proposes mitigation techniques to counter the security threats. The simulation work demonstrates the ability of malicious user to manipulate signals in such a way to confuse signal classifier. The signal classifier is forced by the malicious user to draw incorrect decision boundaries by presenting signal features which are akin to a primary user. Hence, a malicious user is able to classify itself as a primary user and thus gains unrivaled access to the spectrum. First, performance of various classification algorithms are evaluated. K-means and weighted classification algorithms are selected because of their robustness against proposed attacks as compared to other classification algorithm. Second, connection attack, point cluster attack, and random noise attack are shown to have an adverse effect on classification algorithms. In the end, some mitigation techniques are proposed to counter the effect of these attacks.

Spectrum Sensing Security in Cognitive Radio Networks

by

Awais Khawar

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2010

Advisory Committee:
Professor Charles Clancy, Chair/Advisor
Professor Christopher Davis
Professor Charles Silio

Dedication

This thesis is dedicated to my dad, mom, sister, and brother.

Acknowledgments

First and foremost, I am ever grateful to the almighty Allah for his blessings and for having made this work possible.

I owe my gratitude to all the people who have made this thesis possible and because of whom my graduate experience has been one that I will cherish forever. Most importantly, I'd like to thank my advisor Dr. Charles Clancy for giving me an invaluable opportunity to work on an emerging and extremely interesting field of spectrum sensing security in cognitive radio networks. He has always made himself available for help and advice. It has been a pleasure to work with and learn from such an extraordinary individual. Special thanks are due to Professor Christopher Davis and Professor Charles Silio for agreeing to serve on my thesis committee and for sparing their invaluable time reviewing the manuscript and suggestions.

I owe my deepest thanks to my family, my father and mother, who have always stood by me and guided me through my career. They have been a source of inspiration for me. My house mates at my place of residence have been a crucial factor in my finishing smoothly. I'd like to express my gratitude to Akbar, Hamza, Shafiq, Tamoor, and Waseem for their friendship and support.

I would like to acknowledge financial support from the NWFP University of Engineering and Technology, Peshawar, Pakistan for sponsoring my graduate studies at the University of Maryland.

Table of Contents

List of Figures vi

List of Tables vii

1 Introduction 1
- 1.1 Spectrum Reform 1
- 1.2 President's Broadband Policy 3
- 1.3 Military/Defense Requirements 4
- 1.4 Motivation 5

2 Background 6
- 2.1 Software Defined Radios 7
  - 2.1.1 Architecture of SDR 8
- 2.2 Cognitive Radios 11
  - 2.2.1 Cognitive Radio Architecture 12
- 2.3 Cognitive Radio Networks 13
- 2.4 Machine Learning 15
  - 2.4.1 Supervised Learning 15
  - 2.4.2 Reinforcement Learning 16
  - 2.4.3 Unsupervised Learning 17
  - 2.4.4 Applications of Machine Learning for Spectrum Sensing 18
- 2.5 Dynamic Spectrum Access 19
  - 2.5.1 Dynamic Exclusive Use Model 20
  - 2.5.2 Open Sharing Model 21
  - 2.5.3 Hierarchical Access Model 21

3 Spectrum Sensing 23
- 3.1 Challenges 23
- 3.2 Types of Spectrum Sensing 26
  - 3.2.1 Cooperative Sensing 27
  - 3.2.2 Centralized Sensing 27
  - 3.2.3 Distributed Sensing 27
  - 3.2.4 External Sensing 28
- 3.3 How to Sense? 28
  - 3.3.1 Transmitter-based Sensing 29
    - 3.3.1.1 Matched Filtering 29
    - 3.3.1.2 Energy Detector Based Sensing 31
    - 3.3.1.3 Cyclostationary-Based Sensing 33
    - 3.3.1.4 Other Methods 36
  - 3.3.2 Machine Learning-Based Sensing 37
    - 3.3.2.1 Feature Based Signal Classification 37
    - 3.3.2.2 Classification Using Self Organizing Maps 41

4 Security in Cognitive Radios 46
4.1 Building Blocks of Communication Security 47
4.1.1 Availability 47
4.1.2 Integrity 48
4.1.3 Identification 48
4.1.4 Authentication 49
4.1.5 Authorization 49
4.1.6 Confidentiality 49
4.1.7 Non-repudiation 50
4.2 Spectrum Sensing Security 50
4.2.1 Primary User Emulation 51
4.2.2 Attacks Against Cooperative Sensing 54
4.2.3 Threats Against Self-Coexistence Mechanisms 54
4.3 Other Security Threats in CR Networks 55

5 Security Threats in Machine Learning based Spectrum Sensing 56
5.1 Threats to Self Organizing Maps 56
5.1.1 Analytical Derivation 57
5.2 Simulation Scenario 58
5.3 Monte Carlo Simulation 60
5.3.1 Performance of Signal Classifiers 63
5.3.2 No Attack 64
5.3.3 Connection Attack 68
5.3.4 Point Cluster Attack 70
5.3.5 Random Noise Attack 71
5.3.6 Effect of Chaff Points 73
5.4 Real Signals: An Example 75

6 Mitigation and Conclusion 80
6.1 Mitigation 80
6.2 Conclusion 81

Bibliography 82

List of Figures

1.1 Radio spectrum licensed to traditional communication systems 2

2.1 A Typical SDR Receiver. 8
2.2 Cognitive radio architecture showing interaction between software radio, knowledge base, and policy and learning engines. 12
2.3 A Taxonomy of Dynamic Spectrum Access. 20

3.1 Various Aspects of Spectrum Sensing for CR. 24
3.2 Schematic representation of the match filter detector. 30
3.3 Schematic representation of the energy detector over a spectrum band of interest. 32
3.4 Signal Feature Detector 38

5.1 (a) Weight vectors in a 3-dimensional weight space. (b) Neuron densities in a 3-dimensional density space. 61
5.2 Weight vectors and neuron densities for adjacent adversaries without an attack present 66
5.3 Weight vectors and neuron densities for equilateral adversaries without an attack present 67
5.4 (a) Connection attack when adjacent adversarial user adds chaff signals. (b) Neuron density map after the connection attack. 69
5.5 (a) Connection attack when equilateral adversarial user adds chaff signals. (b) Neuron density map after the connection attack. 69
5.6 Weight vectors and neuron densities for adjacent adversaries under the point cluster attack 71
5.7 Weight vectors and neuron densities for equilateral adversaries under the point cluster attack 72
5.8 Weight vectors and neuron densities for adjacent adversaries under the random noise attack 74
5.9 Weight vectors and neuron densities for equilateral adversaries under the random noise attack 74
5.10 Weight vector (neuron) map for BPSK, 16QAM and FM signals . . . 76
5.11 Density map for BPSK and 16QAM signals (notice the clear decision boundary between primary and secondary users) 77
5.12 PSK signal is now a separate class, discernible from FM/QAM class . 78
5.13 Chaff signals connecting two different signal classes i-e FM and 16-QAM signals, making them as one separate class in the density map . 79

List of Tables

3.1 Examples Set of Signal Classification Feature 39

5.1 Performance of Classification Algorithms under No Attack with equilateral adversaries 64

5.2 Performance of Classification Algorithms under No Attack with adjacent adversaries 65

5.3 Adv→Pri error rates for different attack densities using hierarchical clustering with equilateral adversaries 66

5.4 Error types and rates for different attack types using K-means clustering with adjacent adversaries 67

5.5 Error types and rates for different attack types using K-means clustering with equilateral adversaries 70

5.6 Error types and rates for different attack types using hierarchical clustering with adjacent adversaries 73

5.7 Error types and rates for different attack types using hierarchical clustering with equilateral adversaries 75

Chapter 1

Introduction

With the growth of wireless communication technologies the competition for access to electromagnetic spectrum has increased. In order to use the electromagnetic spectrum in an efficient manner a new spectrum sharing technique known as Dynamic Spectrum Access (DSA) was proposed [1]. DSA aims to dynamically allocate spectrum for efficient utilization. This is done by sensing the frequency band for possible white spaces (unoccupied spectrum). The field of spectrum sensing has grown significantly over the past five years, with the growth of cognitive radio technology. Spectrum sensing is required for DSA, spectral awareness, interoperability, and many other smart radio applications [2].

1.1 Spectrum Reform

The radio spectrum is very limited as compared to ever increasing bandwidth requirements of communication technologies. Moreover, in order to keep the transmission interference-free, an exclusive spectrum band is alloted per user which results in an inefficient use of spectrum. The inefficiency is due to the underutilization of spectrum. Some traditional occupants of spectrum are shown in Figure 1.1 [3]. Most of these occupants do not use the spectrum all the time, so if the idle spectrum can be alloted to any other user it can result in greater spectrum efficiency.

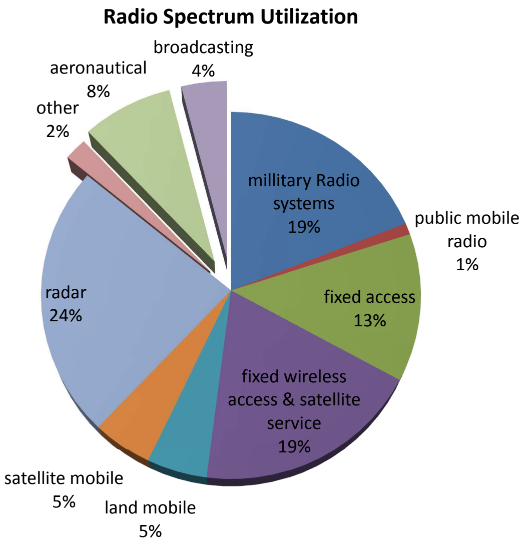

Figure 1.1: Radio spectrum licensed to traditional communication systems

Internationally frequency allocation or licensing is done by the International Telecommunication Union (ITU) which is the United Nations (UN) agency for information and communication technologies. The radio communication sector for ITU (ITU-R) is responsible for economical, efficient, and rational use of spectrum all over the world. ITU-R sets guidelines that helps countries regulate their spectrum. However, nations have the liberty for spectrum use within their boundaries. In the United States of America (USA), spectrum is regulated by Federal Communications Commission (FCC) and National Telecommunication and Information Administration (NTIA). The FCC, which is an independent regulatory agency, administers spectrum for non-Federal use (i.e. state, local government, commercial, private internal business, and personal use) and the NTIA, which is an operating unit of the Department of Commerce, administers spectrum for Federal use (e.g.

use by the Military, the Federal Aviation Authority (FAA), and the Federal Bureau of Investigation (FBI)). The *Spectrum Policy Task Group* of the FCC has set three guidelines for the efficient spectrum utilization [4]:

a. improve access in space, time, and frequency;

b. enable flexible regulation in permitting controlled access to license spectrum, and

c. stimulate efficient spectrum usage through policies.

Possible solutions to the problem of spectrum scarcity is cognitive radios and dynamic spectrum access. It can accommodate ever increasing data rich wireless communication systems by efficient utilization of spectrum. Cognitive radios would autonomously regulate the spectrum with the help of dynamic spectrum access.

1.2 President's Broadband Policy

Historically, the FCC's approach to allocating spectrum has been to formulate policy on a band-by-band, service-by-service basis, typically in response to specific requests for service allocations or station assignments. This approach has been criticized for being ad-hoc, overly prescriptive, and unresponsive to changing market needs. The new *National Broadband Plan* aims to open up vast tracts of underutilized spectrum, not only for licensing by auction, but also for shared and unlicensed use. The plan aims to reallocate 500 megahertz (MHz) of wireless spectrum in the next ten years for mobile, fixed, and unlicensed broadband, of which 300 MHz between 225 MHz and 3.7 gigahertz (GHz) should be made newly available for mobile use within five years [5]. This spectrum would be made available for a variety of licensed and unlicensed flexible commercial use, as well as to meet the broadband needs of specialized users such as public safety, emergency, educational, and other important users. It is notable that the plan recommends the allocation of a new contiguous band of unlicensed spectrum, as well as the rapid implementation of unlicensed access to the unused TV channels known as white spaces [5].

1.3 Military/Defense Requirements

The military is one of the first adopters of cognitive radio technology, with the launch of the Defense Advanced Research Projects Agency (DARPA) Next Generation Communication (XG) program [6]. However, there are many open issues still to be addressed. The XG program goals are to develop both the enabling technologies and system concepts to dynamically redistribute allocated spectrum along with novel waveforms in order to provide dramatic improvements in assured military communications in support of a full range of worldwide deployments [6]. US forces face unique spectrum access issues in each country in which they operate, due to competing civilian or government users of national spectrum. The XG program approach is to develop the theoretical underpinnings for dynamic control of the spectrum, and the technologies and subsystems that enable reallocation of the spectrum. The proposed program goals are to develop, integrate, and evaluate the technology to enable equipment to automatically select spectrum and operating modes to both minimize disruption of existing users, and to ensure operation of US systems. The result of the XG program will be to develop and demonstrate a set of standard dynamic spectrum adaption technologies for legacy and future emitter systems for joint service utility [6].

One of the problems of significant importance is security in cognitive radio systems. Considering the military use of this system the security aspect becomes even more important.

1.4 Motivation

Numerous techniques have been developed to efficiently detect and classify signals in DSA environments. A major area of study has been the use of neural networks for classifying features extracted from signals. Previous work shows the usefulness of machine learning to cognitive radio in signal classification [7, 8]. Research presented in this thesis points out key security issues related to the use of machine learning in cognitive radio networks, specifically related to spectrum sensing attacks, and attacks that can fool signal classifiers. Unsupervised learning is powerful in the sense that minimal preconfiguration is required and radios can learn the properties of other devices in their environment. However, an adversary can also learn the properties of the network thus compromising the security of the network. This thesis explores attacks against signal classifiers and their mitigation techniques in an unsupervised learning environment.

Chapter 2

Background

The field of radio architecture has recently undergone a revolution of design, significantly enabled by Moore's law of computational evolution, where sufficient computational resources are available in Digital Signal Processors (DSPs) and General Purpose Processors (GPPs) to implement the modulation and demodulation, and all the signaling protocols of a radio as a software application. With the exponential growth in the ways and means by which people need to communicate (data communications, voice communications, video communications, broadcast messaging, command and control communications, and emergency response communications) modifying radio devices easily and cost-effectively has become business critical. Commercial wireless communication industry is currently facing problems due to constant evolution of link-layer protocol standards (e.g. 2.5G, 3G, 4G, and beyond) and existence of incompatible wireless network technologies (e.g. WiFi, WiMax, IEEE 802.22, and others) in different countries inhibiting deployment of global roaming facilities. This has led to problems in rolling-out new services/features due to wide-spread presence of legacy subscriber handsets. Software Defined Radios (SDRs), or its more advanced form Cognitive Radios (CRs), offer solution to this problem. CR technology, in which one radio or even a network of radios are able to learn a successful degree of adaptability, that aids the user, the network, and/or the spectrum owner. As new services are offered, more spectrum will be needed. CR will provide the means for radios to communicate with greater spectrum efficiency.

This chapter introduces Software Defined Radios (SDRs) and their typical architecture; Cognitive Radios (CRs), their architecture, and network(s) composed of CRs; Machine Learning (ML) and their usefulness in spectrum sensing for CRs; and Dynamic Spectrum Access (DSA).

2.1 Software Defined Radios

A Software Defined Radio (SDR) is a general-purpose transceiver which supports multiple air interfaces, protocols, coding, and modulation schemes. Moreover, it is reconfigurable via software which runs on Field Programmable Gate Arrays (FPGA), Application Specific Integrated Circuits (ASIC), GPPs, or DSPs [9]. SDRs have transformed future wireless communication devices. In the past, traditional hardware-dependent communication devices had to be changed/upgraded from scratch for new technology as they offered no software control. They were fixed in functions and control. By contrast, SDR technology provides an efficient and comparatively inexpensive solution to this problem. SDR allows multi-mode, multi-band, and/or multi-functional wireless devices that can be enhanced using software upgrades. Though the term SDR was first used by Joseph Mitola [10], it has been in use in the defense sector since the 1970's. The early 1990's saw rapid growth in SDR technology with the Department of Defense (DoD) launching

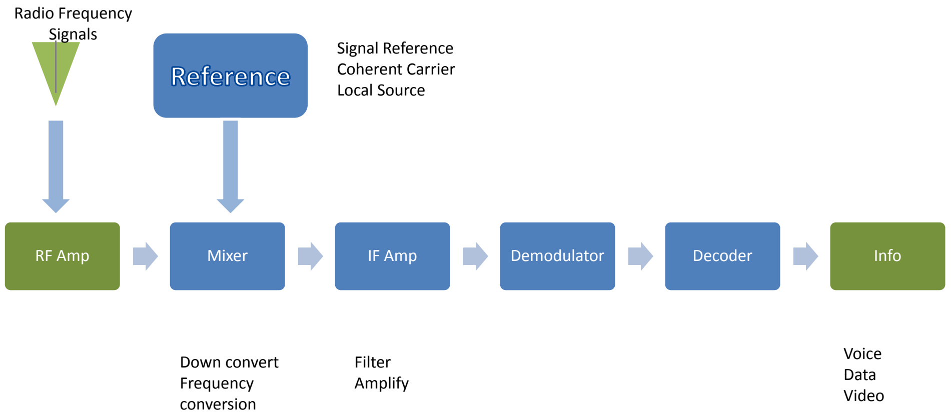

Figure 2.1: A Typical SDR Receiver.

a program called SPEAKeasy [11]. With the success of SPEAKeasy this technology gradually morphed from Speakeasy to Programmable Modular Communication Systems (PMCS), to Digital Modular Radio (DMR) [12], and to the Joint Tactical Radio Systems (JTRS) [13, 14]. SDRs have significant utility for the military and commercial cell phone services, both of which must serve a wide variety of changing radio protocols in real time. SDR technology brings the flexibility, cost efficiency, and power to drive communications forward. SDR has wide-reaching benefits realized by service providers, product developers, and end users [13].

2.1.1 Architecture of SDR

Figure 2.1 shows a typical SDR receiver. Some of the basic building blocks of any SDR are its antenna(s), duplexer and diplexer, radio-frequency (RF) filter, low noise amplifier (LNA), image reject and intermediate-frequency (IF) filter, RF mixer, local oscillator (LO), automatic gain control (AGC), analog-to-digital (ADC) and digital-to-analog (DAC) converters. These building blocks are further discussed in the following sections:

- **Antennas:** Antenna design and selection is very crucial for any wireless device. For a SDR, which is designed to support multiple bands, the antenna choice becomes even more critical. So, an antenna which is capable of operating over a large band is desired for SDR [13].

- **Duplexer and Diplexer:** A duplexer is used to separate transmitted and received signals in a common frequency range that uses a common antenna. Where as, the diplexer isolates the transmitted and received signals in distinct frequency ranges. SDR that support multiple modes, such as full-duplex and half-duplex systems, require a duplexer or diplexer that works for both systems, which is a significant design challenge [13].

- **RF Filter:** This initial filter after the duplexer rejects out-of-band interference. It can also help isolate the receiver from the transmitted signal. This filter should have low loss and provide as much selectivity, as feasible, as possible without limiting the bandwidth needed to support multiple modes of the SDR [13].

- **Low Noise Amplifier (LNA):** The LNA boosts the signal power level into a range compatible with other components in the circuit. The primary design challenge is to maximize gain without adding excessive noise into signal, but this must be traded for power consumption and dynamic range [13].

- **Image Reject and IF filter:** The image reject filter reduces noise and protects the mixer from interference, including any signals located at the image frequency, which, after conversion, may lie in the same band as the desired signal [13].

- **RF Mixer:** The RF mixer is used to down-convert signal and can be a major source of inter-modulation distortion since it is, by its very nature, a non-linear device. Increasing the local oscillator power to the mixer is one way to improve linearity and to reduce distortion, but it reduces the battery life of portable devices. Active RF mixers can also be a source of noise [13].

- **Local Oscillator (LO):** The mixer is driven by a LO whose frequency determines the channel selection. This LO should have a good tuning range and good phase stability to minimize the contributions of phase noise to the noise floor. Thermal noise will also contribute to the noise floor. Power consumption can be a major design issue for LO [13].

- **Automatic Gain Control (AGC):** The AGC is primarily used to ensure that signal has a voltage level that is compatible and makes best use of the input range of ADC. The AGC should be fast enough to account for changing signal levels. It ensures that minimal noise is injected into the system and signal is not clipped by the ADC, which would create non-linear distortion. An AGC should have fast response in order to handle rapid variations of signal levels in situations where fast channel fading is present [13].

- **ADC Converter:** The ADC must sample a real signal at a rate that is at least twice the bandwidth of the signal, and in the case of multi-mode receivers, the highest bandwidth signal dictates sampling rate. Moreover, different signaling standards require different amounts of dynamic range. This makes the design of ADC for SDR more challenging [13].

2.2 Cognitive Radios

Cognitive Radio (CR) is an emerging technology to realize wireless devices with cognition capabilities such as learning, sensing, awareness, and reasoning. They offer global seamless connectivity and solve the interoperability issues among various wireless standards. CRs are a more advanced form of SDRs. They are built on SDR platforms but with additional intelligence. In fact, SDR is a key enabling technology to realize a CR. FCC defines CR as "*A radio or system that senses its operational electromagnetic environment and can dynamically and autonomously adjust its radio operating parameters to modify system operation, such as maximize throughput, mitigate interference, facilitate interoperability, and access secondary markets*" [15]. Hence, they are context aware, they sense and adapt to an ever-changing communication environment. CRs can generally be classified as policy-based or learning radios [8]. In policy based CR, we observe the environment and by reasoning determine how to optimally operate, for example, by switching modulation scheme if signal strength increases/decreases [16]. Learning CRs usually are equipped with Artificial Intelligence (AI), the form of a learning engine. The learning engine uses

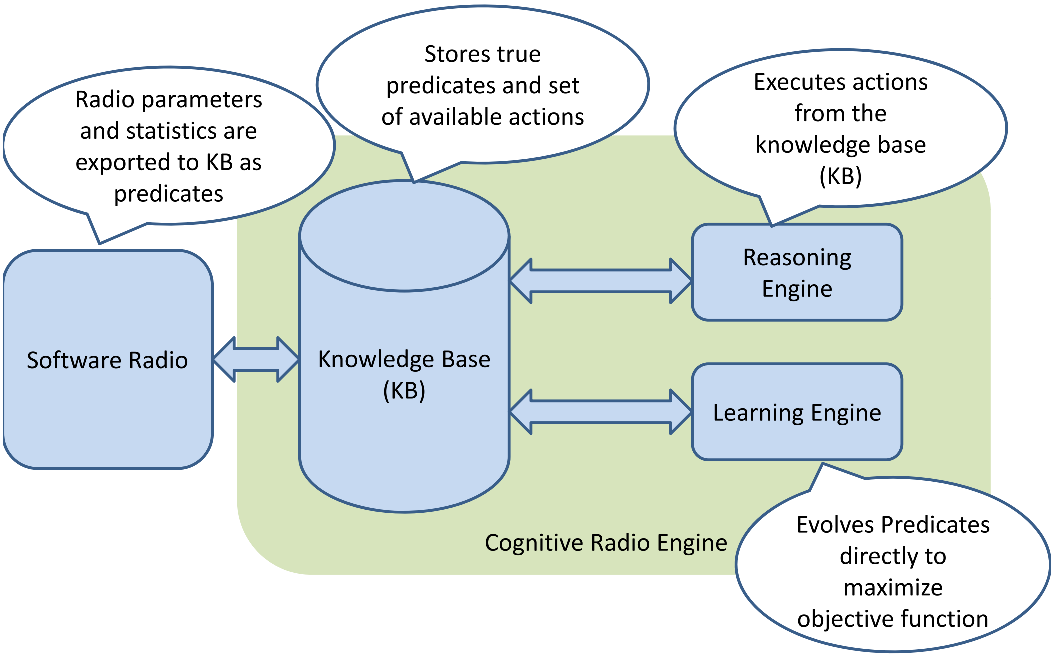
Stores true predicates and set of available actions
Radio parameters and statistics are exported to KB as predicates
Executes actions from the knowledge base (KB)
Software Radio
Knowledge Base (KB)
Reasoning Engine
Learning Engine
Cognitive Radio Engine
Evolves Predicates directly to maximize objective function

Figure 2.2: Cognitive radio architecture showing interaction between software radio, knowledge base, and policy and learning engines.

a knowledge base from experience, and is not dependent on hard rules for decision making as in policy based radios [8]. The learning algorithms could be based on hidden Markov models [17], neural networks [18], or genetic algorithms [19].

Some commercially available CR hardware and software platforms are: GNU Radio [20], Universal Software Peripheral Radio (USRP) [21], and XG Radio by Shared Spectrum [22].

2.2.1 Cognitive Radio Architecture

Figure 2.2 [8], shows a generic CR architecture and the interaction between various components. CR is often thought as an extension to SDR. It adds additional functionalities to SDR like cognition, knowledge base, reasoning engine, and learning engine. A well defined application programming interface (API) dictates communication between the cognitive engine and the software radio. The software radio exports variables that are either read-only or read-write. The read-only parameters represent statistics maintained by the software radio, such as signal-to-noise ration or bit error rate. The read-write variables represent configurable parameters such as transmit power, coding rate, or symbol constellation [8]. Knowledge base is radio's long term memory and helps cognitive engine generate conclusions based on information defined in it. The learning engine is responsible for manipulating the knowledge base from experience. The reasoning engine uses planning, which is a field of AI that works with logic [8].

2.3 Cognitive Radio Networks

Current communication networks limit network's ability to adapt, often resulting in sub-optimal performance. The common network elements (consisting of nodes, protocol layers, policies, and behaviors) are unable to make optimal adaptations with changing network conditions thus resulting in poor performance. Cognitive networks promise to remove these limitations by allowing networks to observe, act, and learn in order to optimize the performance. Cognitive networks are defined by [23] as "*... a network with a cognitive process that can perceive current network conditions, and then plan, decide and act on those conditions. The network can learn from these adaptations and use them to make future decisions, all while tak-*

ing into account end-to-end goals." Cognitive networks are different from cognitive radios in many ways. Cognitive networks aim for end-to-end network performance, whereas cognitive radio's goals are localized only to the radio's user. Here, end-to-end denotes all the network elements involved in the transmission of a data flow.

A cognitive network should provide, over an extended period of time, better end-to-end performance than a non-cognitive network. Cognition could be used to improve resource management, quality of service (QoS), security, access control, and/or many other network goals. Cognitive networks are only limited by the adaptability of the underlying network elements and the flexibility of the cognitive framework. In this manner, cognitive networks are not limited to only wireless networks. Ad-hoc networks, infrastructure-mode wireless networks, fully wired networks and heterogeneous networks are also candidates for cognitive network design [23]. The cost associated with rolling out a cognitive network must be outweighed by the performance improvement the cognitive network provides. The end-to-end goals are what gives a cognitive network its network-wide scope, separating it from other technologies, which have only a local, single element scope [2].

Cognitive networks require a Software Adaptable Network (SAN) to implement the actual network functionality and allow the cognitive process to adapt the network. Similar to a cognitive radio, which depends on an SDR to modify aspects of radio operation (e.g. time, frequency, bandwidth, code, spatiality, and waveform), a SAN depends on a network that has one or more adjustable elements [23]. Practically, this means that a network may be able to modify one or several layers of the network stack in its member nodes. A simple example of a SAN could be a wireless network with directional antennas (antennas with the ability to scan their receive or transmit strength to various points of rotation). A more complex example would involve more modifiable aspects at various layers of protocol stack, such as routing control or Medium Access Control (MAC) [23].

2.4 Machine Learning

In a machine learning environment, a machine changes its structure, program or data (based on its inputs or in response to external observations) in such a manner that its expected future performance improves. It refers to changes in systems that perform tasks associated with AI. Commonly, machine learning systems are classified on the basis of the *underlying learning strategies* used. Camastra and Vinciarelli in [24] identify four different learning types: rote learning, learning from instruction, learning by analogy, and learning from examples. The focus in this thesis is machine learning from examples. Given a set of examples of a concept, the learner induces a general concept description that describe the examples. The three main ways to learn from examples are: *supervised learning*, *reinforcement learning*, and *unsupervised learning* [24].

2.4.1 Supervised Learning

In *Supervised Learning*, the data is a sample of input-output patterns often called the *training sample* or *training set.* The task is to find a deterministic function that maps any input to an output that can predict future input-output observations, and minimize the errors as much as possible. Examples of this learning tasks are the recognition of handwritten letters and digits, the prediction of stock markets and many more [24]. Supervised learning can further be distinguished in *classification* learning and *regression* learning depending on the output. In classification learning, each element of output space is called a class. The output space has no structure except whether two elements of the output are equal or not. On the other hand, if the output space is formed by the values of continuous variables then the learning task is known as the problem of *regression* or *function learning* [25]. Typical examples of regression are to predict the values of shares in the stock exchange market and to estimate the values of physical measure (e.g. temperature) in a section of a thermoelectric plant.

2.4.2 Reinforcement Learning

Reinforcement learning has its roots in control theory. It considers the scenario of a dynamic environment that results in state-action-reward triples as the data. The difference between reinforcement learning and supervised learning is that in reinforcement learning no optimal action exists in a given state, but the learning algorithm must identify an action in order to maximize the expected reward over time. The concise description of data is the strategy that maximizes the reward. The problem of reinforcement learning is to learn what to do, i.e. how to map situations to actions, in order to maximize a given reward. Unlike a supervised learning task, the learning algorithm is not told which actions to take in a given situation. Instead, the learner is assumed to gain information about the actions taken by some reward not necessarily arriving immediately after the action is taken. An example of such a problem is learning to play chess. A comprehensive survey on reinforcement learning can be found in [26].

2.4.3 Unsupervised Learning

If the data is only a sample of objects without associated target values, the problem is known as *unsupervised learning.* In unsupervised learning there is no teacher. Here a concise description of the data can be a set of clusters or a probability density stating how likely it is to observe a certain object in the future. A general way to represent data is to specify a similarity between any pair of objects. If two objects share much structure, it should be possible to reproduce data from the same prototype. This idea underlies *clustering algorithms* that form a rich subclass of unsupervised algorithms. The clustering algorithms aim to find grouping of the objects such that similar objects belong to the same cluster while keeping the the number of clusters fixed. Typical examples of unsupervised learning tasks include the problem of image and text segmentation, and the task of novelty detection in process control [24].

In addition to clustering algorithms, unsupervised learning techniques have algorithms whose aim is to represent high-dimensionality data in low-dimension spaces, trying to preserve the original information of data. These techniques are often called *feature selection* or *dimensionality reduction methods* [24]. The use of more dimensions than necessary leads to several problems. The first one is the space needed to store the data. The speed of algorithms using the data depends on the dimension of the vectors, so a reduction in dimensionality can result in reduced computation time. An example of feature selection algorithm is *Self Organizing Maps* [27].

2.4.4 Applications of Machine Learning for Spectrum Sensing

Traditional spectrum sensing techniques are computationally complex and require significant amount of observation time for adequate performance [7]. Machine learning is a powerful tool which when used in conjunction with CR has promising results for DSA [8]. Signal classifiers can be designed which can do most of the work offline thus reducing online computation. This results in better and quick sensing decisions. Recently, researchers have used supervised, unsupervised, and reinforcement learning for spectrum sensing. Supervised learning requires prior training in order to accurately classify the signals. The supervised learning include the K-nearest neighbor algorithm [28], support vector machines [29], and neural networks [30]. Neural networks have long been considered for pattern recognition and signal classification [30], and have proven to be robust to a variety of conditions such as interfering signals and noise [31]. Researchers have also used *Q-learning* algorithm of reinforcement learning for spectrum sensing. *Mo Lin et al* in [32] showed the use of *Q-learning* for spectrum sensing without the use of any channel state information for estimation of primary traffic. *Reddy* in [33] has discussed the use of *Q-learning* for detection of known and unknown signals.

This thesis demonstrates the use of unsupervised machine learning for spectrum sensing for the first time [34, 35]. Unlike supervised learning, unsupervised learning does not need any training data for signal classification. The examples of unsupervised learning include K-means clustering [36] and self-organizing maps [27]. The applications of machine learning especially the unsupervised learning for Cognitive Radios and DSA are further discussed in Chapter 3.

2.5 Dynamic Spectrum Access

Data transmitted through a wireless channel is not limited to voice any more. With the growth and development of multimedia rich content, the demand for additional spectrum has increased. This has given rise to the problem of spectrum scarcity. But actually, at any given time and location, much of the prized spectrum lies idle. This paradox indicates that spectrum shortage results from the spectrum management policy rather than the physical scarcity of usable frequencies. Traditional approaches of static spectrum allocation are becoming obsolete with the growth in demand of spectrum for high date rate applications. A possible solution to this problem is dynamically sharing spectrum among many users. The utilization of spectrum by different users can be put in two categories: a user having higher preference called as the *Primary User*, and other users wishing to opportunistically access the spectrum called *Secondary Users.* Secondary users have less preference over spectrum than the primary user. A primary user could be the one who owns

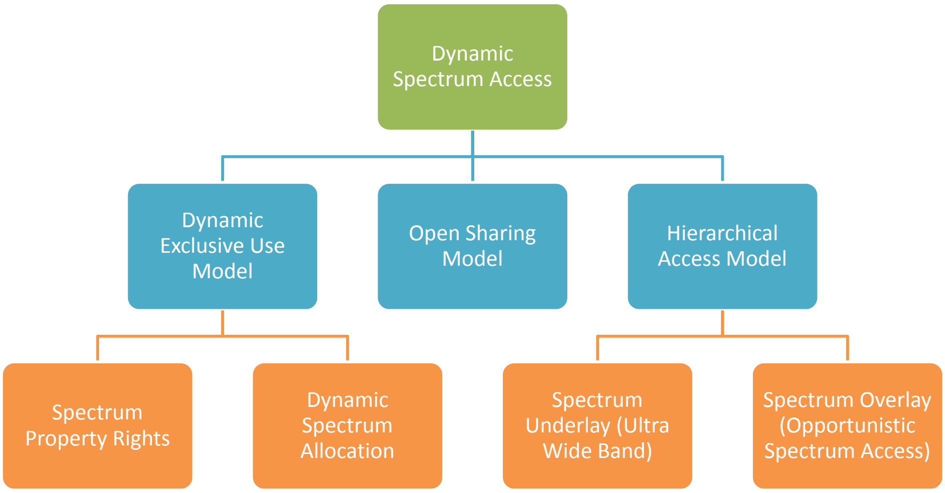
Dynamic Spectrum Access
Dynamic Exclusive Use Model
Open Sharing Model
Hierarchical Access Model
Spectrum Property Rights
Dynamic Spectrum Allocation
Spectrum Underlay (Ultra Wide Band)
Spectrum Overlay (Opportunistic Spectrum Access)

Figure 2.3: A Taxonomy of Dynamic Spectrum Access.

the spectrum and leases out the spectrum to any secondary user when it is unused, given that when the need arises to use the spectrum the secondary user has to vacate the spectrum.

One of the key advantages of CR is that they are not dependent on any fixed license band, they can be reconfigured to any available frequency band. Thus CR offers a solution to efficient utilization of scarce spectrum by dynamically allocating the spectrum. The DSA strategies can be broadly categorized under three models according to Figure 2.3 [37], which are discussed in the following sections.

2.5.1 Dynamic Exclusive Use Model

This model allows the spectrum band to be licensed to services for exclusive use. The main idea is to introduce flexibility to improve spectrum efficiency. The two approaches proposed under this model are: Spectrum property rights [38] and dynamic spectrum allocation [39]. The former approach allows licensees to sell and trade spectrum and to freely choose technology. The second approach, dynamic spectrum allocation, was brought forth by the European DRiVE project [39]. It aims to improve spectrum efficiency through dynamic spectrum assignment by exploiting the spatial and temporal traffic statistics of different services. In other words, in a given region and at a given time, spectrum is allocated to services for exclusive use. Based on an exclusive-use model, these approaches cannot eliminate white space in spectrum resulting from the bursty nature of wireless traffic [37].

2.5.2 Open Sharing Model

This model employs open sharing of spectrum among users in a spectral region. It is also known as *spectral commons* [40]. This spectrum management model supports centralized [41] and distributed [42] sensing. This model is highly successful for wireless services operating in the unlicensed industrial, scientific, and medical (ISM) radio band (e.g., WiFi).

2.5.3 Hierarchical Access Model

This model opens licensed spectrum to secondary users while limiting interference perceived by primary users. This is done on a hierarchical basis. This model supports two approaches for spectrum sharing between primary and secondary user. They are *spectrum underlay* and *spectrum overlay* [37]. The spectrum underlay approach imposes severe constraints on the transmission power of secondary users so that they operate below the noise floor of primary users. By spreading transmitted signals over a ultra-wide frequency band (UWB), secondary users can potentially achieve short-range high data rate with extremely low transmission power. Based on a worst-case assumption that primary users transmit all the time, this approach does not rely on detection and exploitation of spectrum white space. Spectrum overlay was first proposed by Mitola [43] under the term spectrum pooling and then investigated by the DARPA Next Generation (XG) [6] program under the term opportunistic spectrum access. Differing from spectrum underlay, this approach does not necessarily impose severe restrictions on the transmission power of secondary users, but rather on when and where they may transmit. It directly targets at spatial and temporal spectrum white space by allowing secondary users to identify and exploit local and instantaneous spectrum availability in a non-intrusive manner [37].

Compared to the dynamic exclusive use and open sharing models, this hierarchical model is perhaps the most compatible with the current spectrum management policies and legacy wireless systems [37].

Chapter 3

Spectrum Sensing

One of the primary requirement of CRs is their ability to scan the entire band for the presence/absence of primary users. This process is called *Spectrum Sensing*. It is performed locally by a secondary user or collectively by a group of secondary users. Spectrum Sensing is characterized by the join of a quantitative and qualitative analysis of a reference band through the collection of information in terms of frequency usage and air interface classification at a used frequency [44]. The aim is to identify idle spectrum bands/slots known as white spaces. The available spectrum bands are then analyzed to determine their suitability for communication in-terms of signal-to-noise ratio (SNR), error rate, delays, interference, and fading. This chapter discusses types of spectrum sensing, challenges associated in sensing the spectrum, some common spectrum sensing methods, and security in spectrum sensing.

3.1 Challenges

The task of sensing the spectrum accurately and efficiently comes with many challenges, which are discussed in this section.

A. *Hardware requirements*:

The CR should support high sampling rate, large dynamic range for the analog

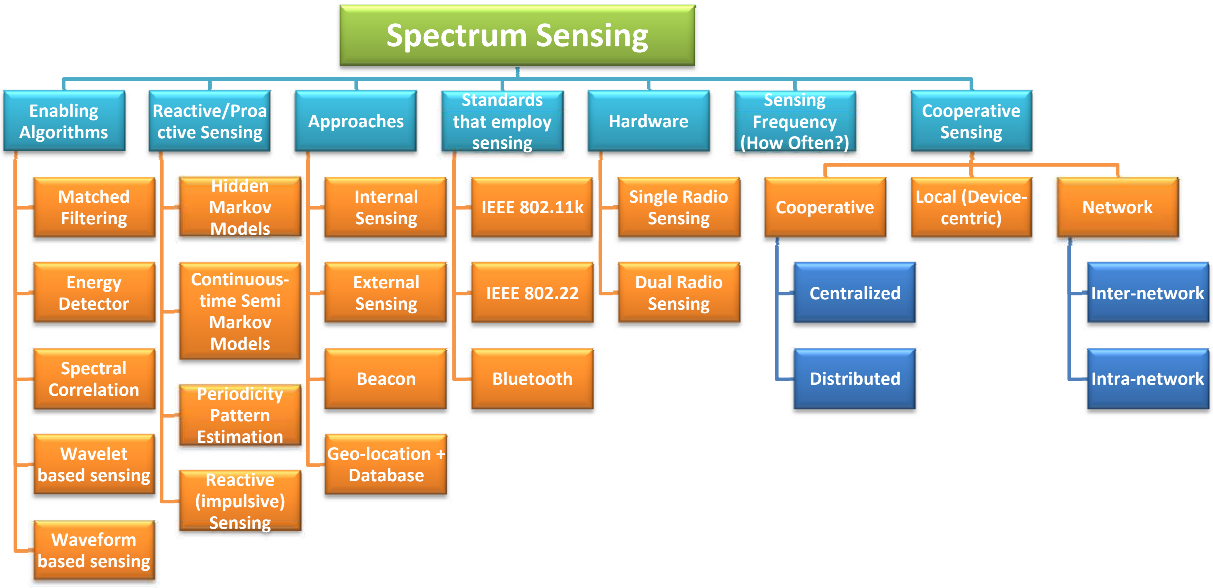

Figure 3.1: Various Aspects of Spectrum Sensing for CR.

to digital converters, high speed signal processor, and multiple analog front end circuitry. Unlike traditional receivers, the CR terminals are required to process transmission over a much wider radio frequency (RF) band. This imposes further constraints on the design of antennas and power amplifiers. In order to keep the delay factor as small as possible, high speed processing units (DSPs or FPGAs) are required to execute computationally demanding signal processing tasks. Two commonly used architectures for spectrum sensing are: *single-radio* and *dual-radio* [45, 46]. The single-radio architecture allots a specific time slot for spectrum sensing. This architecture is very low cost and simple to implement. However, due to the limited sensing duration, accuracy can not be guaranteed. Also, since some time slots are used for sensing rather than sending data, this approach is not spectrally efficient [47, 48]. On the other hand, dual-radio architecture offers separate channels for data transmission and spectrum sensing [49, 50]. Both tasks are executed simultaneously. This approach offers better spectrum efficiency but more power is consumed and expensive hardware is required.

B. *Hidden Primary User*

The hidden primary user problem is similar to the hidden node problem (occurs when a node is visible from a wireless access point (AP), but not from other nodes communicating with said AP) in Carrier Sense Multiple Access (CSMA). Some possible factors leading to this problem are severe multipath fading observed by secondary users while scanning for primary users' transmission. This problem can be solved by using cooperative sensing [51].

C. *Spread Spectrum Users*:

If the primary user is a Code Division Multiple Access (CDMA) device (or any spread spectrum user), its frequency is spread over wide area and its challenging to predict user/detect its frequency if its frequency hop pattern and synchronization scheme is not known before hand [52].

D. *Sensing Duration and Frequency*:

The primary user should be detected as soon as it wants to occupy the channel otherwise it would result in interference. The secondary user must detect the presence of primary user in a timely fashion and thus vacate the spectrum in order to avoid interference. Hence, sensing methods should be able to identify presence of primary user within a certain duration. This requirement poses a limit on the performance of sensing algorithm and creates a challenge for CR design. The effect of sensing time on the performance of CR is investigated in [53]. Sensing frequency, i.e. how often CR should perform spectrum sensing, is a design parameter that needs to be chosen carefully depending upon the temporal characteristics of primary user in the environment [54].

E. *Other factors*:

In addition to above challenges, spectrum sensing algorithms have to be robust, secure, resource efficient, simple to implement, able to support multiple secondary users, power efficient, and resilient to multipath fading [2].

3.2 Types of Spectrum Sensing

The radio spectrum can be sensed by either one cognitive radio or different radios can collaborate and share the information about the spectrum utilization. This approach to sense the spectrum can solve the problem of sensing time, shadowing, noise uncertainty, probability of miss classification, and the hidden primary user problem which are very common in a spectrum sensing environment [51, 55, 56]. Some of the challenges in this approach is to come up with efficient algorithms which are good at sharing the information in a time efficient manner, have reduced complexity and the problem of having a control channel [57]. The following sections discuss various ways to sense the spectrum along with their advantages and disadvantages.

3.2.1 Cooperative Sensing

In cooperative sensing architectures, the control channel can be implemented using different methodologies. These include a dedicated band, unlicensed band (such as Industrial, Scientific and Medical (ISM)), and underlay Ultra Wideband (UWB) system [58]. One of these methods can be selected depending on the system requirements. The shared spectrum decisions can be soft or hard decisions made by each CR [59]. Cooperative sensing can be implemented in two fashions: *centralized* or *distributed* [1]. These methods are explained in the next two sections.

3.2.2 Centralized Sensing

In centralized sensing, a central agent is given the task of handling and dispatching knowledge of spectrum, i.e. identification of available spectrum, which it gets from other participating CRs. The hard (binary) sensing results are gathered at a central place which is known as Access Point (AP) in [55]. The main disadvantage is dependency on a single centralized agent, any problem with this can easily mess up the whole system. On the other hand, centralized sensing promises considerable performance gains over other sensing methods [60].

3.2.3 Distributed Sensing

In distributed sensing, CRs share information among themselves but when it comes to decisions they make their own decisions as to which part of spectrum to occupy/vacate. In application scenarios involving geographically distributed radios, such as a wireless communication system, distributed spectrum sensing approaches are worth considering due to the variability of the radio signal [46, 52, 55]. Such methods may significantly increase the reliability of the spectrum estimation process, at the expense of computational complexity and power/bandwidth usage for the transmission of spectrum sensing information. The main advantage comes from the fact that no central authority is required for making decisions and thus has reduced cost. A distributed CR architecture for spectrum sensing is given by [61] and various algorithms are discussed in [2]. The results show that the performance of CR improves considerably through collaborative sensing [62].

3.2.4 External Sensing

In external sensing, an external agent performs the sensing and broadcasts the channel occupancy information to cognitive radios. The main advantages are overcoming hidden primary user problem and uncertainty due to shadowing or fading. The spectrum efficiency is increased as the CR don't spend time for sensing. Also, it is power efficient as the sensing terminal need not be mobile and not necessarily powered by batteries [2]. External sensing is one of the methods proposed for identifying primary users in IEEE 802.22 standard [45].

3.3 How to Sense?

There are numerous methods proposed by various researchers on how to sense the spectrum. The following sections first discuss some previous techniques used for spectrum sensing. Then spectrum sensing using machine learning, specifically unsupervised learning, is explored in great detail.

3.3.1 Transmitter-based Sensing

Transmitter detection approach is based on the detection of the weak signal from a primary transmitter through the local observations of secondary users. Three schemes are generally used for the transmitter detection. In the following subsections, matched filter detection, energy detection, and cyclo-stationary feature detection techniques are discussed.

3.3.1.1 Matched Filtering

A matched filter is an optimum receiver for an AWGN channel [63]. In cognitive radios the matched filter is also an optimum method for detection of primary user when transmitted signal is known [63]. However, one requires exact information about signal transmitted, such as its bandwidth, operating frequency, and modulation type. With the advantage of having the short time to achieve probability of miss detection [64] the main disadvantage is its implementation complexity [52] and before-hand knowledge of signal. This is still possible since most primary users have pilots, preambles, synchronization words or spreading codes that can be used for coherent detection [52]. Some examples are: TV signal has narrow-band pilot for audio and video carriers; CDMA systems have dedicated spreading codes for pilot and synchronization channels; OFDM packets have preambles for packet acquisition

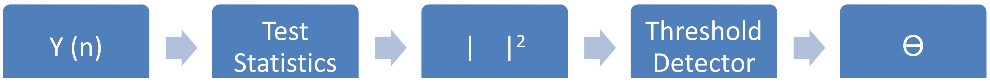

Figure 3.2: Schematic representation of the match filter detector.

[52]. Another disadvantage of match filtering is large power consumption as various receiver algorithms needs to be executed.

A typical match filter detector for spectrum sensing is shown in Figure 3.2 [65]. Let the sample received signal $y(n)$ at the CR user be

$$y(n) = \theta h p(n) + w(n), \quad 0 \leq n \leq N-1 \tag{3.1}$$

where $p(n)$ denote the pilot sequence, $w(n)$ denote the white noise, h denote the quasi-static block fading channel from the primary transmitter to the CR user, and $\theta = 0$ and $\theta = 1$ denote the absence and presence of the primary signal, respectively. Define

$$P_p = \frac{1}{N} \sum_{n=0}^{N-1} |p(n)|^2 \tag{3.2}$$

as the average power of the pilot signal. Then the instantaneous SNR within the current detection period is given by

$$\gamma = \frac{|h|^2 P_p}{\sigma_n^2} \tag{3.3}$$

where σ_n^2 denotes the noise power.

$$Y = \left| \sqrt{\frac{2}{N P_p \sigma_n^2}} \sum_{n=0}^{N-1} y(n) p^*(n) \right|^2 \tag{3.4}$$

$$Y = \begin{cases} \left| \sqrt{\frac{2}{NP_p\sigma_n^2}} \sum_{n=0}^{N-1} w(n)p^*(n) \right|^2, & H_0 \\ \left| \sqrt{\frac{2NP_p}{\sigma_n^2}} h + \sqrt{\frac{2}{NP_p\sigma_n^2}} \sum_{n=0}^{N-1} y(n)p^*(n) \right|^2, & H_1 \end{cases} \tag{3.5}$$

and makes decision accordingly as

$$\theta = \begin{cases} H_1, & \text{if Y} > \lambda \\ H_0, & \text{if Y} < \lambda \end{cases} \tag{3.6}$$

where the threshold λ is chosen to satisfy a target false probability. In order to keep computation simple and mathematically tractable an AWGN channel is selected. Under this assumption, it is shown in [65], the test statistics of the MF detector Y follows a central chi-square distribution with two degree of freedom under H_0 and a non-central chi-square distribution with two degree of freedom and a non-centrality parameter $\mu = 2N\gamma$ under H_1, i.e.

$$f_y(Y) \sim \begin{cases} \chi_2^2, & H_0 \\ \chi_2^2(\mu), & H_1 \end{cases} \tag{3.7}$$

based on which the false-alarm probability and the detection probability for a given threshold can be obtained [65].

3.3.1.2 Energy Detector Based Sensing

A simple approach as compared to match filtering is to perform non-coherent detection through energy detection (ED). The signal is detected by comparing the output of energy detector with a threshold which depends on noise floor. It is one

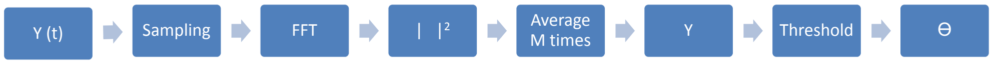

Figure 3.3: Schematic representation of the energy detector over a spectrum band of interest.

of the most common way to sense the spectrum because of its low implementation and computational complexities. Also, the receiver does not need to know any thing about the received signal's characteristics.

There are several drawbacks of energy detectors that might diminish their simplicity in implementation. First, a threshold used for primary user detection is highly susceptible to unknown or changing noise levels. Second, energy detector does not differentiate between modulated signals, noise and interference. Since, it cannot recognize the interference, it cannot benefit from adaptive signal processing for canceling the interferer. Lastly, an energy detector does not work for spread spectrum signals: direct sequence and frequency hopping signals, for which more sophisticated signal processing algorithms need to be devised [52].

To detect the primary signal by the energy detector in AWGN channel is to distinguish between the following hypotheses:

$$y(t) = \begin{cases} i(t) + w(t), & H_0 \qquad Signal Absent \\ s(t) + i(t) + w(t), & H_1 \qquad Signal Present \end{cases} \tag{3.8}$$

where $y(t)$ is the received signal at the cognitive radio, $s(t)$ is the transmitted signal from the primary transmitter, $i(t)$ is interference, and $w(t)$ is the additive white Gaussian noise (AWGN). H_0 and H_1 denote the hypothesis corresponding to the absence and presence of the primary signal, respectively [65]. An energy detector can be implemented as in Figure 3.3 [65, 66]. The spectral component on each spectrum subband of interest is obtained from the fast Fourier transform (FFT) of the sampled received signal. Then the test statistics of the ED is obtained as the observed energy summation within M consecutive segments, i.e.,

$$Y = \begin{cases} \sum_{m=1}^{M} |W(m)|^2, & H_0 \\ \sum_{m=1}^{M} |S(m) + W(m)|^2, & H_1 \end{cases} \tag{3.9}$$

where $S(m)$ and $W(m)$ denote the spectral components of the recieved primary signal and the white noise on the subband of interest in the mth segment, respectively. The decision of the ED regarding the subband of interest is given by

$$\theta = \begin{cases} H_1, & \text{if Y} > \lambda \\ H_0, & \text{if Y} < \lambda \end{cases} \tag{3.10}$$

where the threshold λ is chosen to satisfy a target false alarm probability. Once λ is determined, the detection probability can be obtained [65]. In ED, processing gain is proportional to FFT size N and observation/averaging time T. Increasing N improves frequency resolution which helps narrow band signal detection. Also, longer averaging time reduces the noise power thus improves SNR.

3.3.1.3 Cyclostationary-Based Sensing

Modulated signals are in general coupled with sine wave carriers, pulse trains, repeating spreading, hoping sequences, or cyclic prefixes which result in built-in periodicity. Even though the data is a stationary random process, these modulated signals are characterized as cyclostationary, since their statistics, mean and autocorrelation, exhibit periodicity. This periodicity is typically introduced intentionally in the signal format so that a receiver can exploit it for: parameter estimation such as carrier phase, pulse timing, or direction of arrival. This can then be used for detection of a random signal with a particular modulation type in a background of noise and other modulated signals.

Common analysis of stationary random signals is based on autocorrelation function and power spectral density. On the other hand, cyclostationary signals exhibit correlation between widely separated spectral components due to spectral redundancy caused by periodicity [67]. The distinctive character of spectral redundancy makes signal selectivity possible. Signal analysis in cyclic spectrum domain preserves phase and frequency information related to timing parameters in modulated signals [67]. As a result, overlapping features in the power spectrum density are non overlapping feature in the cyclic spectrum. Different types of modulated signals (such as BPSK, QPSK, SQPSK) that have identical power spectral density functions can have highly distinct spectral correlation functions [52]. Furthermore, stationary noise and interference exhibit no spectral correlation, thus differentiating modulated signals, interference and noise in low signal to noise ratios [52].

Mathematically cyclostationary detection is realized by analyzing the cyclic autocorrelation function (CAF) [68] of the received signal, or, equivalently, its two-dimensional spectrum correlation function (SCF) [69] since the spectrum redundancy caused by periodicity in the modulated signal results in correlation between widely separated frequency components [69]. Consider a typical digitally modulated signal of the form

$$s(t) = \sum_n a(n)g(t - nT_o - t_o) \tag{3.11}$$

where T_o is the symbol period, t_o is an unknown timing offset, and $g(t)$ is the shaping pulse. For simplicity, assume that the sequence $a(n)$ is stationary with zero mean and variance σ_a^2; then the time-varying autocorrelation function (TVAF) of $s(t)$ is defined as

$$\begin{aligned} R_s(t,\tau) &= E[s(t+\tau)s^*(t)] \\ &= \sum_n \sigma_a^2 g(t+\tau-nT_o-t_o)g^*(t-nT_o-t_o) \\ &= \sum_{\alpha=\frac{k}{T_o}} R^\alpha(\tau)\exp^{\jmath 2\pi\alpha t} \end{aligned} \tag{3.12}$$

where

$$R^\alpha(\tau) = \begin{cases} \dfrac{\sigma_a^2 \exp^{\jmath 2\pi\alpha t_o}}{T_o} \displaystyle\int G^*(f+\alpha)G(f)\exp^{\jmath 2\pi f\tau}\,df, & \alpha = \dfrac{k}{T_o} \\ 0, & otherwise \end{cases} \tag{3.13}$$

and $G(f)$ is the Fourier transform of $g(t)$ [65].

The function $R^\alpha(\tau)$ is called the CAF and α is called cyclic frequency. As indicated in [70], the CAF at a given cyclic frequency α determines the correlation between spectral components of the signal separated in frequency by an amount of α. In general [69], the CAF of cyclostationary signals is nonzero only for integer multiples of a fundamental cyclic frequency α_o. For the signal model given in [71], $\alpha_o = 1/T_o$. Thus, given T_o, the CAF can be utilized to determine the presence or absence of the primary signal by evaluating the values of $R^\alpha(\tau)$ at corresponding cyclic frequencies [65]. In addition to continuous time domain detection, cyclostationary detection can also be implemented in discrete time domain [65, 68]. Some common cyclic frequencies for signals of practical interest are [72, 73]:

1. *Analog TV signal*: It has cyclic frequencies at multiples of the TV-signal horizontal line-scan rate (15.75 KHz in USA, 15.625 KHz in Europe).

2. *AM signal*: $x(t) = a(t)cos(2\pi f_c t + \phi_o)$. It has cyclic frequencies at $\pm 2f_c$.

3. *Pulse modulated (PM)* and *frequency modulated (FM)* signal:

   $x(t) = a(t)cos(2\pi f_c t + \phi(t))$. It usually has cyclic frequencies at $\pm 2f_c$. The characteristics of the spectral-correlation density (SCD) function at cyclic frequency $\pm 2f_c$ depend on $\phi(t)$.

4. Digital-modulated signals are as follows:

   (a) *Amplitude-Shift Keying*: $x(t) = \sum_{n=-\infty}^{\infty} a_n p(t - n\triangle_s - t_o)cos(2\pi f_c t + \phi_o)$. It has cyclic frequencies at $k/\triangle_s$, $k \neq 0$ and $\pm 2f_c + k/\triangle_s$, $k = 0, \pm 1, \pm 2, \ldots$

   (b) *Phase-Shift Keying*: $x(t) = cos[2\pi f_c t + \sum_{n=-\infty}^{\infty} a_n p(t - n\triangle_s - t_o)]$. For BPSK, it has cyclic frequencies $k/\triangle_s$, $k \neq 0$ and $\pm 2f_c + k/\triangle_s$, $k = 0, \pm 1, \pm 2, \ldots$ For QPSK, it has cycle frequencies at $k/\triangle_s$, $k \neq 0$.

3.3.1.4 Other Methods

Some other proposed spectrum sensing methods in the literature are based on waveform sensing [74], radio identification [75], multitaper spectral estimation [76], wavelet transform based spectral estimation [77], Hough transform [78], neural networks [7], statistical features based sensing [79], and time-frequency analysis of the received signal [44].

3.3.2 Machine Learning-Based Sensing

Spectrum sensing techniques based on machine learning have shown promising results over traditional spectrum sensing methods. Classification algorithms based on machine learning techniques fall into one of to major categories: supervised learning and unsupervised learning. In supervised learning, training data is fed to the classifier a priori and the training data is annotated as to the class it falls into. While this type of training often yields the most robust results, access to such training data can often be impractical. On the other hand, unsupervised learning does not need any training phase. By learning from signals encountered on the fly, initial classification results can be more error prone, but as more and more examples are seen, the classification engine can leverage the wealth of observed data to robustly classify new signals. To work well, the signal statistics need to be linearly separable to facilitate use of various clustering algorithms. Machine learning based sensing techniques are discussed in the following sub sections.

3.3.2.1 Feature Based Signal Classification

Consider a system of machine learning where a series of signal values $x_n(t)$ are presented to a signal classifier, whose goal is to determine whether $x_n(t)$ is a primary user P or secondary user S. If $x_n(t)$ is a primary user, then the band it occupies,

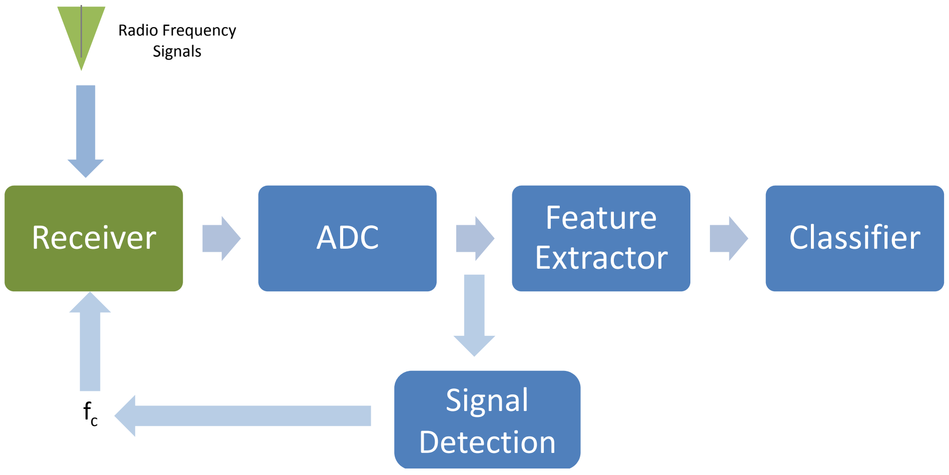

Figure 3.4: Signal Feature Detector

and possibly even several adjacent bands, is off-limits to use by secondary users. If $x_n(t)$ is a secondary user, then these bands may be shared by other secondary users. This thesis looks at approaches that find the answer to this question in an unsupervised manner, or without external expert help. It is these approaches that are more flexible and in turn have more value as a signal classification technique.

Figure 3.4 shows a typical signal feature extractor. Signal features when combined with SDR enables DSA applications. One can look at either the temporal or spectral, or both, characteristics of the signal. Some of the signal features of interest are given in the Table 3.1 [80]. These features can be extracted through the analysis of cyclostationary features or the power spectral density (PSD) of the signals. The Fast Fourier Transform (FFT) can be used to transform the time-domain signal into the frequency domain and thus the PSD of the signal can be estimated. PSD can then be used to extract first-order spectral features such as the bandwidth, received power, and the roll-off factor [80].

Table 3.1: Examples Set of Signal Classification Feature

Signal Feature	**Feature Type**
Received Signal Power	Spectral
Baseband Bandwidth	Spectral
Roll-off Factor	Spectral
Alpha Profile	Spectral
Standard Deviation of Amplitude	Temporal
Standard Deviation of Phase	Temporal
Standard Deviation of Envelope Amplitude	Temporal
Standard Deviation of Change in Phase	Temporal
Standard Deviation of Absolute Value of Change in Phase	Temporal
Nth Order Moment/Cumulant of Amplitude	Temporal
Nth Order Moment/Cumulant of Phase	Temporal

A. Bayesian Theory Approach

Bayesian Theory of Decision (BTD) is a fundamental tool of analysis in machine learning. The fundamental idea in BTD is that the decision problem can be solved using probabilistic considerations. The goal here is to distinguish between two classes: primary P and secondary S. Let $C = [P, S]$ be the set of classes. The correct decision $\hat{C}_n$ is made by maximizing the following conditional probability:

$$\hat{C}_n = \arg \max_{C \in [P,S]} P(C|x_n(t)) \tag{3.14}$$

Using the Bayes rule, this is equivalent to:

$$\hat{C}_n = \arg \max_{C \in [P,S]} \frac{P(x_n(t)|C)P(C)}{P(x_n(t))} \tag{3.15}$$

$$\hat{C}_n = \arg \max_{C \in [P,S]} P(x_n(t)|C)P(C) \tag{3.16}$$

Note that in equation (3.15), $P(x_n(t))$ is not part of maximization and is therefore removed in equation (3.16), it is just a normalization factor ensuring that the sum of probabilities is one. The **a priori** probability $P(C)$ can be computed using prior knowledge of the breakdown between primary and secondary users in a particular frequency band.

The a posteriori probability $P(x_n(t)|C)$ is more difficult to compute. In particular $x_n(t)$ is a vector of large dimension, and the joint probability distribution across an arbitrarily large-dimension $x_n(t)$ and the two classes P and S is difficult to compute.

B. Features Extraction

The a posteriori probability $P(x_n(t)|C)$ can be computed by first projecting $x_n(t)$ into a features space using transform $F : \mathbb{C}^\infty \rightarrow \mathbb{R}^N$. This reduces the dimensionality of the problem by examining specific features of $x_n(t)$ rather than $x_n(t)$ itself. Now, one can use a classification engine to compute likelihoods $L_C(.)$ of the various classes. Specifically,

$$\begin{aligned} L_C(F(x_n(t))) &\propto P(F(x_n(t)|C)) \\ &\approx P(x_n(t)|C) \end{aligned} \tag{3.17}$$

In unsupervised learning, one feeds a series of feature vectors $F_n = F(x_n(t))$ into the classification engine, and it then outputs class P or S based on its acquired knowledge. At no point one provides any annotated training data to the classifier that helps it make decisions about which points belong to which class.

3.3.2.2 Classification Using Self Organizing Maps

Self organizing map (SOM) [27] is a commonly used algorithm for unsupervised learning. Its a type of neural network where individual weights are evolved to fit the input data. With this approach an input vector is presented to the network and the output is compared with the target vector. If they differ, the weights of the network are altered slightly to reduce the error in the output. This is repeated many times and with many sets of vector pairs until the network gives the desired output. A SOM learns to classify the training data without any external supervision whatsoever. A SOM does not need a target output to be specified unlike many other types of network. Instead, where the node weights match the input vector, that area of the lattice is selectively optimized to more closely resemble the data for the class the input vector is a member of. From an initial distribution of random weights, and over many iterations, the SOM eventually settles into a map of stable zones. Each zone is effectively a feature classifier, so each graphical output is a type of feature map of the input space. Any new, previously unseen input vectors presented to the network will stimulate nodes in the zone with similar weight vectors. Training occurs in several steps and over many iterations:

Learning Algorithm:

(a) Each node's weights are initialized. Consider l number of neurons with m dimensional weight vector $\mathbf{w_j} = [w_1, w_2, ..., w_m]$, where $j = 1, 2, ..., l$.

Initialize the parameter t, which is a count of iterations:

$$t = 0$$

(b) Consider an m dimensional input vector $\mathbf{x}$ chosen at random from the training set $\mathcal{X} = [\mathbf{x_1}, ..., \mathbf{x_l}]$ and presented to the lattice.

(c) Every node is examined to calculate which one's weights are most like the input vector. The closest weight vector to input vector, in terms of distance, is commonly known as winning neuron denoted as

$$s(\mathbf{x}) = \arg\min_j \|\mathbf{x} - \mathbf{w_j}\|_2 \tag{3.18}$$

(d) The radius of the neighborhood of the winning neuron is now calculated. This is a value that starts large, typically set to the 'radius' of the lattice, but diminishes at each time-step. So typically each weight vector is adapted according to:

$$\Delta\mathbf{w_r} = \epsilon(t)h(d_1(r,s))(\mathbf{x} - \mathbf{w_r}) \tag{3.19}$$

where:

$$h(d_1(r,s)) = exp(\frac{-d_1(r,s)^2}{2\sigma(t)^2}) \tag{3.20}$$

$$\epsilon(t) = \epsilon_i(\frac{\epsilon_f}{\epsilon_i})^{\frac{t}{t_{max}}} \tag{3.21}$$

$$\sigma(t) = \sigma_i(\frac{\sigma_f}{\sigma_i})^{\frac{t}{t_{max}}} \tag{3.22}$$

In equation (3.20), $h(.)$ is a Gaussian neighborhood function and $d_1(r,s)$ is a function that depends on the Eucledian distance between units r and

s that are images of weight vectors $\mathbf{w_r}$ and $\mathbf{w_s}$ on the grid, respectively. In equation (3.21), ϵ_i and ϵ_f correspond to initial and final values of the learning rate and t_{max} is the total number of iterations. In equation (3.22), $\sigma(t)$ is time varying variance, by time varying it means that $\sigma(t)$ varies with number of iterations, and as the number of iteration increases $\sigma(t)$ decreases and thus the neighborhood shrinks. $\sigma(t)$ is defined in equation (3.22), where σ_i and σ_f correspond to initial and final variance values.

(e) Each neighboring node's (the nodes found in step 4) weights are adjusted to make them more like the input vector. The closer a node is to the winning neuron, the more its weights get altered.

(f) Increase the time parameter t:

$$t = t + 1$$

(g) If $t < t_{max}$ go to step 2 [24, 81].

Imagine neurons n_i located on a lattice in a one or two-dimensional space, called map space. Let n_i represent the location in that space of the neuron. Each neuron has an associated weight vector $w_i \in \mathbb{R}^N$ which is a point in weight space. These neurons serve to map points from the N-dimensional weight space to the low-dimensional map space. For each input feature vector F_n, the first step in classification is to select the neuron with closest weight vector. In particular,

$$\hat{j} = \arg\min_j \|F_n - w_j\|_2 \tag{3.23}$$

Now, weight vectors of all neurons are updated, where the magnitude of the the update is a function of the distance between the neuron and $n_{\hat{j}}$:

$$w_i \leftarrow w_i + \eta_n a_{i,\hat{j}}(F_n - w_i) \tag{3.24}$$

where $a_{i,\hat{j}}$ is an activation metric based either on Euclidean or link distance (e.g. $a_{ii} = 1$, $a_{ij} = 0.5$ if i and j are neighbors, and otherwise $a_{ij} = 0$), and η_n is an exponentially-increasing time metric for some time-constant τ.

$$\eta_n = \eta_o e^{\frac{n}{\tau}} \tag{3.25}$$

The definition of a_{ij} means that weight vector updates will most influence the selected neuron and its neighbors. The definition of η_n means that the more samples fed into the system, the less the system will update its weight vectors. This damping effect allows convergence. The values η_o and τ affect this convergence rate. Initial values for weight vectors w_i can be selected randomly using a uniform distribution, a rough approximation of the weight space probability distribution, or they can be selected based on initial feature vectors. In particular, for this last approach assume feature vectors $F_1, F_2 \cdots F_n$ are known, where $n \geq N$. This can be placed into a matrix $\mathcal{F}$ as follows:

$$\mathcal{F} = \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{bmatrix} \tag{3.26}$$

From $\mathcal{F}$ one can compute its eigenvalues $\lambda_1, \lambda_2, \ldots \lambda_N$ and eigenvectors $e_1, e_2, \ldots, e_N$. Assuming neurons are in a two-dimensional space, and given λ_j and λ_k are the two largest eigenvalues, e_j and e_k are the principle eigenvectors that span a two-dimensional space best fit for the original N-dimensional data. For lattice point n_i one can compute its initial weight vector w_i by taking n_i's two-dimensional coordinate on the plane spanned by e_j and e_k and projecting it into the higher-dimensional $\mathbb{R}^N$ space. If the two signal classes P and S have sufficiently separable signatures in the weight space $\mathbb{R}^N$, the weight vector of map neurons in cluster together after . By looking at weight vector densities across the lattice, the clusters can be identified, and decision boundaries can be placed between them. For each new signal received, the map can be used to classify it, while simultaneously updating itself with the new information.

Application of SOM for Spectrum Sensing:

This algorithm is useful for scanning an entire band of frequency of interest, locating energy, and then using features from that energy to feed into an untrained classifier. Since, this algorithm does not need any training data, the spectrum sensing is performed in a quick and efficient manner.

Chapter 4

Security in Cognitive Radios

Cognitive radio paradigm introduces entirely new classes of security threats and challenges different than those frequently encountered in a traditional wireless network. This is because of additional features the CR offers, such as sensing, geolocation, spectrum management, access to policy database etc. Each of these functions and processes need to be accessed for potential vulnerabilities, and security mechanism needs to be established. Thus, providing security in a CR environment is far more challenging as compared to traditional wireless networks. However, this issue needs to be addressed in order to make CR a long-term commercially-viable concept. This chapter discusses the topics of wireless security in cognitive radio networks, delineating the key challenges in this area. Some of the fundamental questions which arise in CR security are [82]:

(a) What are the potential threats to a cognitive radio network?

(b) What are the potential attacks against cognitive radio network?

(c) What is the likelihood of these threats and attacks?

(d) What is the potential consequence of these attacks?

Some of the fundamental blocks of communication security are availability, integrity, identification, authentication, authorization, confidentiality, and non-repudiation. So besides providing typical traditional forms of security, cognitive networks must provide enhanced security mechanisms for various cognitive functions it supports. As with any other wireless network, the security of CR at each layer is critical. Muhammad in [83] discusses CR security at the physical layer, data-link layer, network layer, transport layer, and application layer. He also discusses some cross-layer attacks against CR networks. In this thesis, the focus is on spectrum sensing security. Ensuring the trustworthiness of spectrum sensing process is an important problem that needs to be addressed.

This chapter first discusses some building blocks of security applicable to cognitive radios, then spectrum sensing security is discussed in detail.

4.1 Building Blocks of Communication Security

In this section, some building blocks of communication security are introduced and how these building blocks are applied in cognitive radio networks are discussed.

4.1.1 Availability

One of the fundamental requirements for any communication device and/or network is its availability for use at all times. Most of the attacks in communications like the denial-of-service (DoS) attacks, jamming attacks, buffer over flow attacks on network queues are all targeted towards rendering the network unavailable either temporarily or permanently [83]. In the case of CRs, network availability means the ability of primary user and secondary user to access the spectrum. For the primary user, availability refers to being able to transmit in the licensed band without interference from the secondary user. On the other hand, availability for secondary users is the existence of chunks of spectrum where it can transmit without causing interference to primary user. Security mechanisms should ensure spectrum availability for both primary and secondary users.

4.1.2 Integrity

Data in a network needs to be protected from malicious modification, insertion, deletion, or replay. Integrity assures that the data received is exactly as sent by an authorized entity. Data integrity is extremely important in wireless networks, where unlike wired networks the wireless medium is easily accessible to intruders. In a CR, data integrity means that only authorized primary and secondary users are able to communicate [83].

4.1.3 Identification

Identification is one of the main security requirements for any communication device. Every device must have a unique identifier. In a CR, the secondary users must have a tamper-proof identification mechanism.

4.1.4 Authentication

Authentication is assurance that the communication entity is the one that it claims to be. The objective of authentication is to prevent unauthorized users from gaining access to the system. From the service provider's perspective, authentication protects the service provider from unauthorized intrusions into the system. In CR networks, there is an inherent requirement to distinguish between primary and secondary users. Therefore, authentication can be considered as one of the basic requirements for cognitive radio networks [83].

4.1.5 Authorization

In DSA environment, secondary users are authorized to use the channel when its not being used by the primary user or if white spaces are available. They can use the spectrum conditioned that they won't cause any interference to the primaries' transmission. If a secondary user (possibly malicious) is causing interference to primary user, it should be stopped. But, it is difficult to pinpoint exactly which secondary user is causing interference and even more so in a distributed setting [83].

4.1.6 Confidentiality

Confidentiality and integrity are linked very closely. While integrity assures that data is not maliciously modified in transit, confidentiality assures that the data is transformed in such a way that it is unintelligible to an unauthorized (possibly malicious) entity. This is commonly done by ciphering and encrypting the data with a secret key which is shared only with the recipient. The error-prone and noisy nature of wireless channel poses a unique challenge to both data confidentiality and integrity, since they rely on ciphers which are sensitive to channel errors and erasures. This issue is even more pronounced in CR networks, where the secondary user's access to network is opportunistic and spectrum availability is not guaranteed [83].

4.1.7 Non-repudiation

Non-repudiation techniques [84] prevent either the sender or receiver from denying a transmitted message. Therefore, when a message is sent, the receiver can prove that a message was in fact sent by the alleged sender. Similarly, when a message is received, the sender can in fact prove that the data received was by the alleged receiver. In CR setting, if malicious secondary users violating the protocols are identified, non-repudiation techniques can be used to prove the misbehavior and disassociate/ban the malicious user from the network [83].

4.2 Spectrum Sensing Security

An important feature of CR is its ability to sense the spectrum. It should be able to distinguish primary user signals from the secondary user signals in a robust way. The secondary users are permitted to operate in the licensed band only on a non-interference basis to primary users. Since wireless communication is of bursty nature, the usage of licensed spectrum bands by primary user may be sporadic. So a CR must constantly monitor for the presence of primary user signals in the current operating band and candidate bands. If a secondary user detects the presence of primary user in the current band it must vacate the band for the primary user and switch to one of the fallow bands. On the other hand, if the secondary user detects the presence of an unlicensed user, it invokes a coexistence mechanism to share spectrum resources [85, 86].

4.2.1 Primary User Emulation

In a hostile wireless environment a malicious user may modify the air interface to mimic a primary user's signal characteristics. This results in secondary users identifying the malicious user as a primary user, thus, vacating occupied spectrum band for the primary user. In this way the malicious user gets unrivaled access to the primary user's spectrum band. In literature this kind of attack against cognitive radio networks is considered as Primary User Emulation (PUE) [87]. The PUE attack can be launched while the spectrum is being sensed by energy detection or cyclostationary signal features are used for primary user's signals.

When *energy detection* is used, a secondary user can recognize other secondary users but is unable to recognize primary user. So, when a secondary user detects a signal that it recognizes, it labels it as of secondary user; otherwise it determines that signal is that of a primary user. The detection of primary user's signal is very simple using this approach. Thus, a selfish or malicious secondary user can easily exploit the spectrum sensing process.

When *cyclostationary feature detection* is used, in which intrinsic characteristics of primary user's signals are used to distinguish from those of secondary users, an attacker can still make its signals indistinguishable from primary user signals by transmitting signals that have same characteristics as primary signals. If an attacker uses such a mechanism, CR that receive the signal will falsely identify the malicious user's (attacker) signal as that of primary user.

The aim of PUE attacks is to disrupt the communication and use the spectrum resources that could have been used by legitimate secondary users. Depending on the motivation behind attack, *Ruiliang et al* in [87] classify them as either a selfish PUE or a malicious PUE attack.

- **Selfish PUE Attacks:** In this attack, an attackers objective is to maximize its own spectrum usage. When selfish PUE attackers detect a fallow spectrum band, they prevent other secondary users from competing for that band by transmitting signals that emulate the signal characteristics of primary user signals. This attack is most likely to be carried out by two selfish secondary users whose intention is to establish a dedicated link [87].

- **Malicious PUE Attacks:** In this attack, the objective is to obstruct

> the DSA process of legitimate secondary users i.e. prevent legitimate secondary users from detecting and using fallow licensed spectrum bands, thus causing *denial of service.* Unlike a selfish attacker, a malicious attacker does not necessarily use fallow spectrum bands for its own communication purposes. It is quite possible for an attacker to simultaneously obstruct the DSA process in multiple bands by exploiting two DSA mechanisms implemented in every CR. The first mechanism requires a CR to wait for a certain amount of time before transmitting in the identified fallow band to make sure that the band is indeed unoccupied. The second mechanism requires a CR to periodically sense the current operating band to detect primary user signals and to immediately switch to another band when such signals are detected. By launching a PUE attack in multiple bands in a round-robin fashion, an attacker can effectively limit the legitimate secondary users from identifying and using fallow spectrum bands [87].

In order to mitigate these attacks against spectrum sensing, *Ruiliang et al* in [87] propose a transmitter verification scheme, called *localization-based defense (LocDef)*, which utilizes both signal characteristics and location of the signal transmitter to verify primary signal transmitters. The localization scheme utilizes an underlying wireless sensor network (WSN) to collect snapshots of received signal strength (RSS) measurements across a CR network. By smoothing the collected RSS measurements and identifying the RSS peaks, transmitter location is estimated.

4.2.2 Attacks Against Cooperative Sensing

Another form of security threat occurs when using cooperative spectrum sensing [55]. Although cooperative spectrum sensing has promising advantages in terms of accurate sensing of the spectrum, but, if a CR reports false information the whole system may be jeopardized in a way that too many incorrect sensing decisions would occur thus increasing the likelihood of false detection. It is challenging to devise robust algorithms for cooperative spectrum sensing which are attack proof.

4.2.3 Threats Against Self-Coexistence Mechanisms

The coexistence between primary (incumbent) users and secondary users is referred to as incumbent coexistence. On the other hand, coexistence between secondary users in different wireless radio access network (WRAN) cells is referred to as self-coexistence [88]. Self-coexistence mechanisms are needed in overlapping coverage areas of CR networks to minimize self interference and utilize spectrum efficiently. Unfortunately, adversaries can modify/forge self-coexistence control packets to exploit self-coexistence mechanisms, which can result in drastic reduction of network capacity. What makes the task of protecting self-coexistence control packets, using conventional cryptosystems, difficult is the need to use an inter-operator key management system. It is likely that the networks that contend for spectrum (via self-coexistence mechanisms) will be managed by different wireless service operators. Designing and maintaining an inter-operator key management system could be complex and expensive [87].

4.3 Other Security Threats in CR Networks

In addition to spectrum access-related security threats, software-centric signal processing by (software-based) CR systems also raises new security implications. For instance, the download process of the radio software needs to be secured. Moreover, the radio software itself needs to be tamper resistant once it is downloaded on the radio terminal so that software changes cannot be made to cause a radio to operate with parameters outside of those that were approved [87].

Chapter 5

Security Threats in Machine Learning based Spectrum Sensing

The DSA is an emerging technology which can solve the problem of spectrum scarcity. Machine learning can be used to improve the performance and robustness of DSA [31]. Previous work shows the sensitivity of cognitive radios against emulation attacks [87, 89]. In supervised or unsupervised machine learning environment a malicious user can manipulate the feature extractors and classifier engines to affect their output and thus results in the misclassification of the intended signals [80, 90]. Chapter 4 discussed importance of security when using machine learning for DSA. Here the focus is on the role of security in an unsupervised machine learning environment and this thesis explores some possible types of security vulnerabilities against signal classifiers.

5.1 Threats to Self Organizing Maps

This section deals with a generic self-organizing map whose goal is to distinguish between two classes. First, analytical derivation of the feasible set of input signals that would inductively manipulate the decision regions is done, and then a more complex example through simulation is demonstrated.

5.1.1 Analytical Derivation

Consider, for simplicity, a one-dimensional map. After $n-1$ iterations from previous input data, the weight vectors for neurons n_i are w_i, respectively, are updated. Our goal is to create an input vector $x_n(t)$ that will cause a neuron currently on the border between two classes to switch classes. Applying this technique inductively, an attacker can arbitrarily shift the decision region between primary and secondary users, causing more signals to be classified as primary, decreasing competition for spectral resources. Lets assume neurons $n_1, \ldots, n_l$ are linearly arranged and evenly spaced, and without loss of generality, $n_i = i$. Assume nodes $n_1, \ldots, n_{l-i}$ are classified as primary users, and nodes $n_i, \ldots, n_l$ are classified as secondary users. Our goal is to cause n_i to be classified as a primary user. To accomplish this, the input signal must have a feature vector closer to w_i than any other weight vector. This introduces our first constraint:

$$\|F(x_n(t)) - w_i\|_2 < \|F(x_n(t)) - w_j\|_2 \, \forall \, j \neq i \tag{5.1}$$

where w_i and w_j are the i-th and j-th weight vectors, respectively. In other words, $x_n(t)$ must have a feature vector that causes n_i to be the winning neuron. Let χ_1 be the space of feasible vectors $x_n(t)$ that satisfy this space. Next, the map will update the weight w_i, and the goal is for this update δ_i to cause neuron n_i to shift from being classified as secondary to primary. However this update also affects n_i's neighbors. For n_i and its neighbors, the update is defined as

$$\delta_j = \eta_n a_{j,i}(F(x_n(t)) - w_j) \tag{5.2}$$

Assuming activation values of 1 for the winning neuron, 0.5 for its neighbors, and 0 for all other neurons, the result is following:

$$\delta_{i-1} = 0.5\eta_n(F(x_n(t)) - w_{i-1}) \tag{5.3}$$

$$\delta_i = \eta_n(F(x_n(t)) - w_i) \tag{5.4}$$

$$\delta_{i+1} = 0.5\eta_n(F(x_n(t)) - w_{i+1}) \tag{5.5}$$

If n_i is on the decision boundary and currently in the same class as n_{i+1}, it must be closer in weight space to n_{i+1} than n_{i-1}. To change this behavior, the new weight vector need to be closer to n_{i-1}. Quantitatively, this means

$$\|(w_i + \delta_i) - (w_{i-1} + \delta_{i-1})\|_2 < \|(w_i + \delta_i) - (w_{i+1} + \delta_{i+1})\|_2 \tag{5.6}$$

substituting equation (5.5) into equation (5.6) results in

$$\begin{aligned} &\|(1 - \eta_n)w_i - (1 - 0.5\eta_n)w_{i-1} + 0.5\eta_n F(x_n(t))\|_2 \\ &\qquad < \|(1 - \eta_n)w_i - (1 - 0.5\eta_n)w_{i+1} + 0.5\eta_n F(x_n(t))\|_2 \end{aligned} \tag{5.7}$$

This is the second constraint. Let χ_2 be the feasible set of signals $x_n(t)$ that satisfy this constraint. If $\chi_1 \cap \chi_2 \neq \phi$, then a signal exists that will alter the decision boundaries.

5.2 Simulation Scenario

The three types of attacks explored in this thesis, connection attack, point-cluster attack, and random noise attack, are defined in the following sections.

(a) Connection Attack

The connection attack aims to link two signal classes together by the use of chaff signals. The purpose is to confuse the signal classifier and force it to draw inaccurate decision boundaries. So chaff points are added which have means collinear with μ_1 and μ_3, and are added at random points in between the two. Their goal is to confuse the classifier and case points with mean μ_3 to be in the same output class as points with mean μ_1, rather than μ_2. This confuses the signal classifier and it draws boundaries showing adversarial user to be the primary user and all others to be secondary user, thus showing that in a self-organizing map the decision boundaries can be manipulated by an adversary if it presents known feature values.

(b) Point Cluster Attack

In the *point cluster* attack many chaff points are added all in one place, the idea is to confuse the signal classifier and force it to draw decision boundaries showing adversarial user to be the primary user. This approach is successful, since in many classification algorithms, say, K-means, the goal is to try to find the center which is the mean position of all the samples in the class. The presence of many chaff points at a particular point drags the mean of adjacent classes closer to itself. In practice if too many signals are produced with the same statistical properties the signal classifier can be forced to term this one as a single independent class and all other classes to be a second class. Thus an adversary producing large amount of signals can learn the environment of cognitive radio thus compromising the security of the primary user.

(c) Random Noise Attack

In the *random noise* attack chaff points are added randomly all over the map space. The idea is to confuse the signal classifier so that it draws the decision boundaries randomly, because after the training phase the the randomly added chaff signals will try to get close to the already present signals thus forming small clusters of neurons in the weight space close to them. The randomness in the decision boundaries is in the sense that some times it classifies adversarial user to be the primary user and sometimes it draws boundaries accurately. Through simulation it is shown that this type of attack is also strong enough to confuse the signal classifier and thus, when effective, draws inaccurate decision boundaries.

5.3 Monte Carlo Simulation

This section demonstrates the efficacy of the attack on features generated from real signal data. Signal features which are consistent with real signal, namely standard deviation of time-domain signal, standard deviation of time-averaged time-domain signal, and the standard deviation of the derivative of the signal are used. It is assumed that the primary user is a frequency-modulated (FM) analog signal, secondary users are binary phase-shift keying (BPSK) and ad-

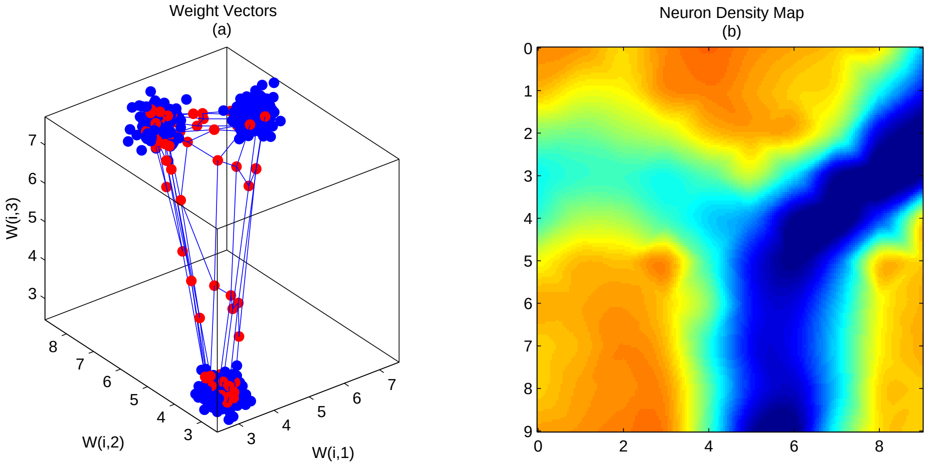

Figure 5.1: (a) Weight vectors in a 3-dimensional weight space. (b) Neuron densities in a 3-dimensional density space.

versary users are 16-point quadrature amplitude modulation (16-QAM) [34]. This thesis starts the discussion by explaining the significance of results in the presence of no attack, connection attack, and point cluster attack in both adjacent and equilateral adversaries case. Finally the random noise attack and its significance in the two different cases of adversarial users is explained. This thesis shows numerically the error rates for adjacent adversaries and equilateral adversaries in the presence of no attack, connection attack, point cluster attack, and the random noise attack. This thesis also demonstrates the relationship between the strength of the attacks on the signal classifier and the number of chaff signals.

In order to demonstrate the efficacy of decision boundary movement for a self organizing map with a two-dimensional map space this thesis uses the

MATLAB Neural Network Toolbox to implement the self-organizing maps. The Monte Carlo simulation executes the following scenario 200 times and averages the final result in order to show misclassification rates. The reason to run the simulation 200 times and not more is that after a certain number of training the weight vectors do not change much and a certain optimal training threshold is achieved.

A network with a two-dimensional map space and three-dimensional weight space is created. The weight space could be arbitrarily large, but for the purposes of visualization, three dimensions were used. The weight space spans values [0, 10] in each dimension. Input data samples are taken from two Gaussian distributions. The means of the two Gaussian distributions are chosen in such a way so that in one set the adversaries are adjacent to each other and in the other they are equidistant.

Adjacent Adversaries:

μ_1 = (3,3,3), μ_2 = (7,7,7), μ_3 = (5,8,7)

Equilateral Adversaries:

μ_1 = (3,3,3), μ_2 = (4,7,5), μ_3 = (5,3,7)

where μ_1, μ_2 and μ_3 correspond to the means of primary, secondary, and adversarial users respectively. The standard deviation is 0.25. The network is first trained to input samples, this case uses training data of 600 samples, and then runs the classification algorithm and determines primary and secondary users through the decision boundary algorithm. The input points are shown as blue dots in Figure 5.1(a). The network is trained to the input samples and neuron weights are shown in red. The associated neuron densities are depicted in Figure 5.1(b). There is a clear decision boundary in dark blue that separates the primary from secondary users.

5.3.1 Performance of Signal Classifiers

In an unsupervised learning environment, the signal classifiers are very sensitive to an attack because they update themselves when new data arrives so an adversary can manipulate the output of the classifier in the long run [80]. This thesis used SOM as a signal classifier and K-means and hierarchical clustering algorithms. For K-means, $K = 2$ is used for two classes of primary and secondary users, in the hierarchal algorithm classification on the basis of weighted, average, complete, single and ward algorithms was explored. The 'weighted' uses the weighted average distance, 'average' uses the unweighted average distance, 'complete' uses the furthest or largest distance, 'single' uses the shortest distance, 'ward' uses the inner squared distance among the weight vectors in the two clusters, note that this thesis takes into account only the Euclidean distance and the hierarchical tree structure is monotonic. The performance of these algorithms is evaluated under no attack. Table 5.2 gives the statistics with adjacent adversaries. Notice that the 'single' hierarchical algorithm is more prone to attack since even under no attack it draws incorrect decision boundaries 6% of the time. Table 5.1 evaluates the performance

Table 5.1: Performance of Classification Algorithms under No Attack with equilateral adversaries

Classification Algorithm	Pri→Sec	Sec→Pri	Adv→Pri
K-means	0	0.56	0.30
Weighted	0	0.54	0.44
Average	0	0.84	0.16
Complete	0	0.90	0.10
Single	0	0.68	0.56
Yard	0	0.70	0.30

with equilateral adversaries, keeping in view the average performance of these algorithms under this situation one finds that K-means and 'weighted' perform better and are less prone to attack on the average as compared to other classification algorithms used. This work uses these two classification algorithms, K-means and weighted, in order to evaluate their performance under no attack, connection attack, point cluster attack and random noise attack.

5.3.2 No Attack

Figures 5.2 and 5.3 gives a general overview of how K-means behaves in the case of adjacent adversaries and equilateral adversaries, respectively, with no attack. K-means is able to accurately draw boundaries for the primary and secondary users under no attack. Table 5.4 shows that the K-means signal

Table 5.2: Performance of Classification Algorithms under No Attack with adjacent adversaries

Classification Algorithm	Pri→Sec	Sec→Pri	Adv→Pri
K-means	0	0	0
Weighted	0	0	0
Average	0	0	0
Complete	0	0	0
Single	0	0.06	0.06
Yard	0	0	0

classifier is highly successful in drawing accurate decision boundaries under no attack with adjacent adversaries but with equilateral adversaries the performance of K-means degrades and it classifies adversary user to be primary user 35% of the time (see Table 5.5) [35]. So even under no attack the false positive rate is much high for K-means under the equilateral adversaries case. The weighted hierarchical clustering algorithm performs much like K-means when used with adjacent adversaries (see Table 5.6), but with equilateral adversaries it has a high false positive rate as compared to K-means under similar situation (see Table 5.7) [35].

Table 5.3: Adv→Pri error rates for different attack densities using hierarchical clustering with equilateral adversaries

Attack Type	0	200	400	600
Point Cluster	0	0.91	0.96	0.98
Connectivity	0.44	0.52	0.58	0.81
Random Noise	0.39	0.46	0.50	0.54

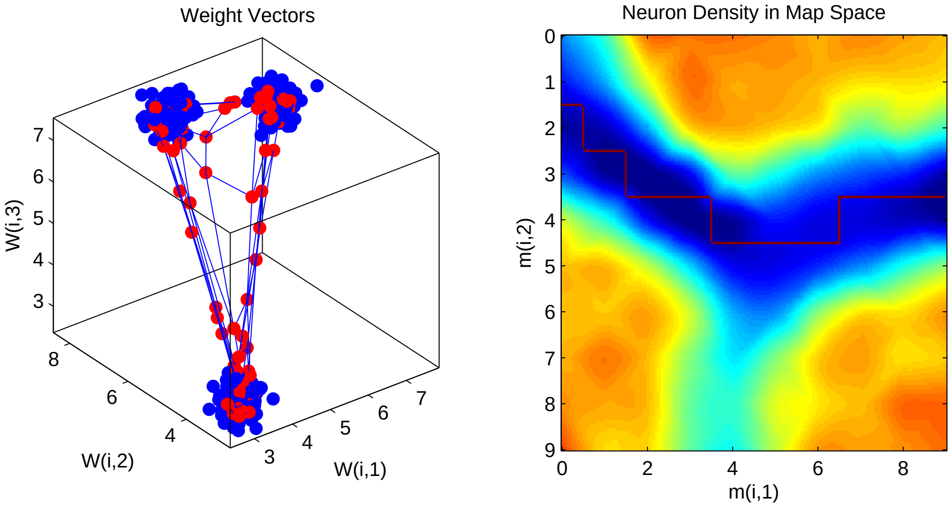

Figure 5.2: Weight vectors and neuron densities for adjacent adversaries without an attack present

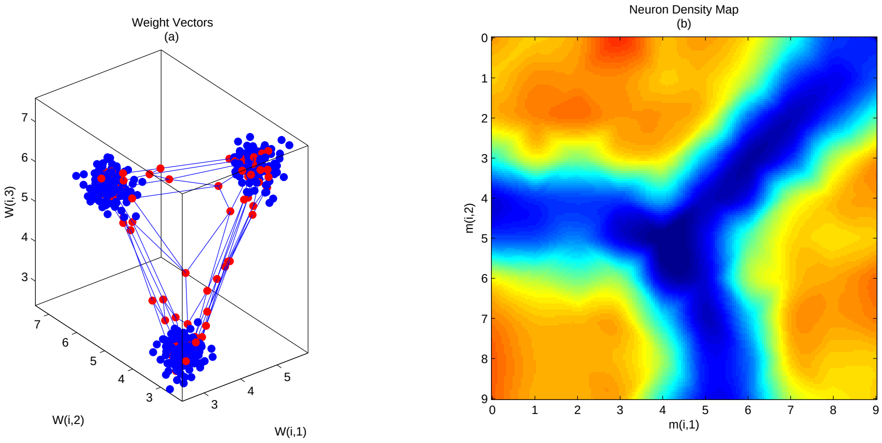

Figure 5.3: Weight vectors and neuron densities for equilateral adversaries without an attack present

Table 5.4: Error types and rates for different attack types using K-means clustering with adjacent adversaries

Error Type	None	Connect	Cluster	Noise
Pri→Sec	0	0	0	0
Sec→Pri	0	0	0.35	0
Adv→Pri	0	0	0.38	0

5.3.3 Connection Attack

The connection attack aims to confuse the signal classifier by presenting signal features similar to that of primary user. This attack uses chaff signals which have means collinear with the means of primary user and adversarial user. Actually it demonstrates that if the adversarial user knows signal features of primary user this type of attack can be launched against signal classifiers [34]. Figures 5.4 and 5.5 show the connection attack in the case of adjacent and equilateral adversaries, respectively [35]. Observe the chaff signals which are added to fool the signal classifier. Compare Figures 5.4(b) and 5.5(b) with Figures 5.2(b) and 5.3(b). Notice the decision boundaries before attack, Figures 5.2(b) and 5.3(b), and after attack, Figures 5.4(b) and 5.5(b). The alteration in the decision boundaries of primary and adversary user shows the efficacy of the connection attack. Next, the performance of K-means and weighted algorithms under adjacent and equilateral adversaries is evaluated. Tables 5.4 and 5.6 show that connection attack has no impact on clustering algorithms with adjacent adversaries, however, Tables 5.5 and 5.7 show that with equilateral adversaries the connection attack has high error rate. The connection attack is more powerful if weighted algorithm is used because then it will classify adversary to be primary 75% of the time but with K-means the error rate is 36%. So it makes sense to use that clustering algorithm which is less prone to attakcs on signal classifiers i.e. K-means in this case.

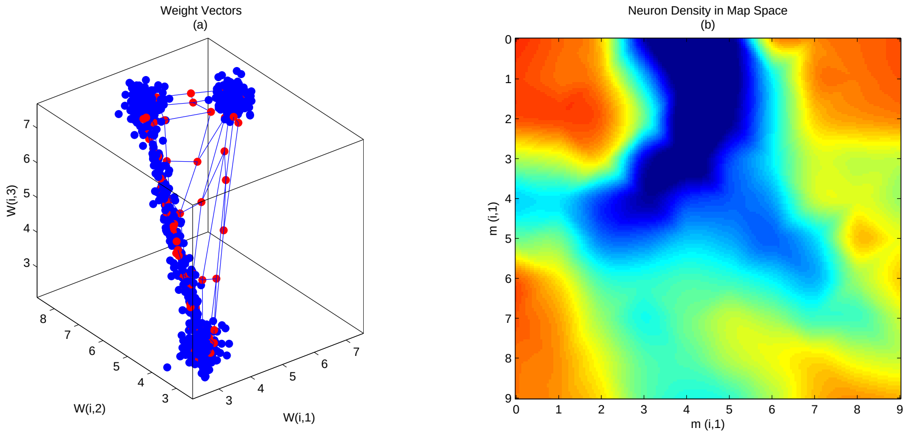

Figure 5.4: (a) Connection attack when adjacent adversarial user adds chaff signals. (b) Neuron density map after the connection attack.

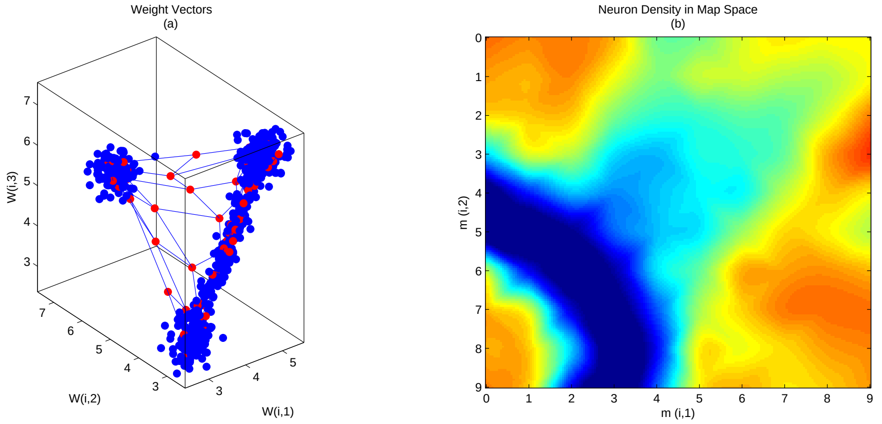

Figure 5.5: (a) Connection attack when equilateral adversarial user adds chaff signals. (b) Neuron density map after the connection attack.

Table 5.5: Error types and rates for different attack types using K-means clustering with equilateral adversaries

Error Type	None	Connect	Cluster	Noise
Pri→Sec	0	0	0	0
Sec→Pri	0.38	0.33	1	0.36
Adv→Pri	0.35	0.36	1	0.58

5.3.4 Point Cluster Attack

The point cluster attack in the adjacent adversaries case is shown in Figure 5.6. Notice the location of the cluster of chaff points. The chaff points are added with mean (9,9,9) and its adjacent signals have means (5,8,7) and (7,7,7), the classification algorithm lumps these three into a single class because of their close proximity where as the signal class with mean (3,3,3) are classified as the other class. The neuron density map has a dense region showing the presence of many chaff points. Figure 5.7 shows the point cluster attack in the equilateral adversaries case. The means of the signals in this case are (9,9,9) for chaff signals, and the other three signals have means (3,3,3), (4,7,5) and (5,3,7). Notice the chaff signals are far from the other signals in the weight vector map as compared to the adjacent adversaries case. Tables 5.4 and 5.6 suggest that weighted algorithm performs better under point cluster attack with adjacent adversaries as the error rate it produces is much lower than K-means, so a weighted clustering algorithm will have better decision power over

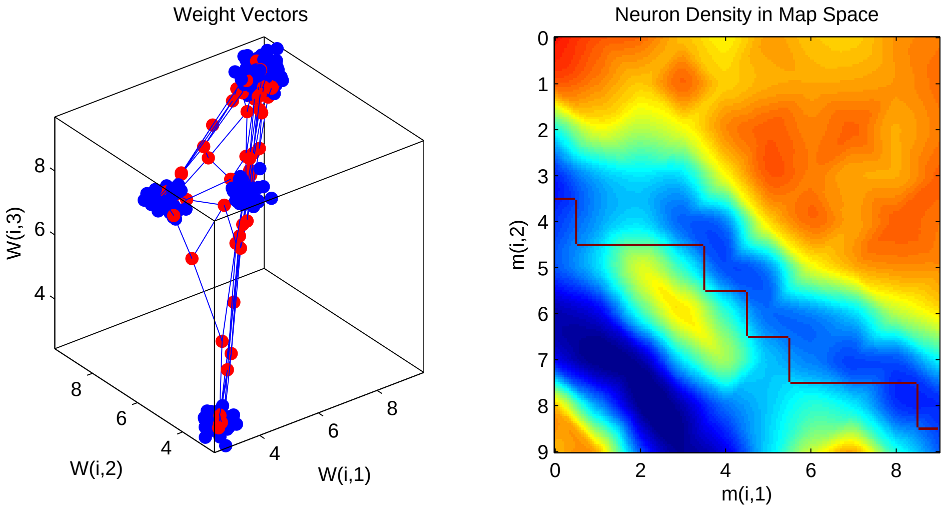

Figure 5.6: Weight vectors and neuron densities for adjacent adversaries under the point cluster attack

K-means under this attack. Also, Tables 5.5 and 5.7 show that the weighted algorithm outperforms K-means with equilateral adversaries. Its interesting to note that K-means completely fails to draw decision boundaries accurately whereas the weighted algorithm is some-what successful but still has a 98% error rate.

5.3.5 Random Noise Attack

The random noise attack case is a special case in which the signals generated have means which are randomly generated using a uniform distribution, in terms of neural networks this corresponds to neurons all over the weight vector space. Due to the random nature of these signals the signal classifier finds it difficult to draw accurate decision boundaries. It is difficult for K-means signal

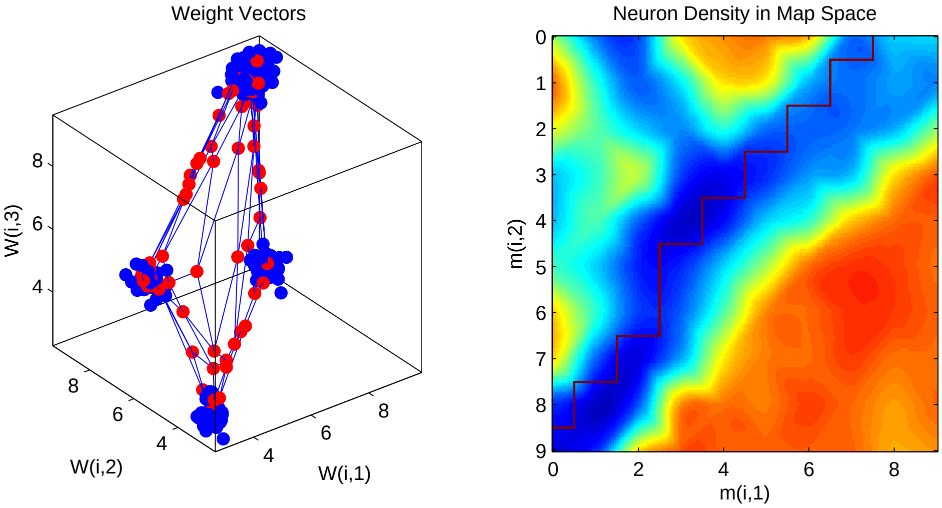

Figure 5.7: Weight vectors and neuron densities for equilateral adversaries under the point cluster attack

classifier to locate the mean of the samples when they are randomly distributed over the space of signals so it draws the decision boundaries arbitrarily.

Figures 5.8 and 5.9 show the weight vectors and the neuron density map when a random noise attack is performed on the adjacent and equilateral adversaries cases respectively. Note that the chaff signals are randomly distributed over the weight space. After the training phase, they get aligned more close to the three classes depending upon the orientation of their weight vectors. The neuron density map shows the two decision regions. Notice that the darker and more dense areas correspond to our signal classes whereas the less dense areas are the result of the random chaff signals located arbitrarily. Every time the decision boundary algorithm runs on this attack it tries to find the best mean taking in account the random chaff signals present around original

Table 5.6: Error types and rates for different attack types using hierarchical clustering with adjacent adversaries

Error Type	None	Connect	Cluster	Noise
Pri→Sec	0	0	0	0
Sec→Pri	0	0	0.10	0.38
Adv→Pri	0	0	0.11	0.39

classes and splits the decision region into primary and secondary user. Table 5.4 shows that the random noise attack has no effect on K-means with adjacent adversaries but if weighted algorithm is used then poor decision regions are obtained as shown by Table 5.6. Table 5.5 shows that in the case of equilateral adversaries this attack is successful on K-means. A comparison of random noise attack in equilateral adversaries case shows that its difficult to conclude whether K-means is a good algorithm for drawing decision boundary for primary and secondary users as compared to the weighted algorithm, since in both cases the error rates are very close.

5.3.6 Effect of Chaff Points

It is interesting to note the behavior of signal classifiers under the point cluster attack and random noise attack with adjacent and equilateral adversaries by changing the number of chaff signals. The intensity of attack increases with the increase of chaff points. This makes sense since by increasing the number

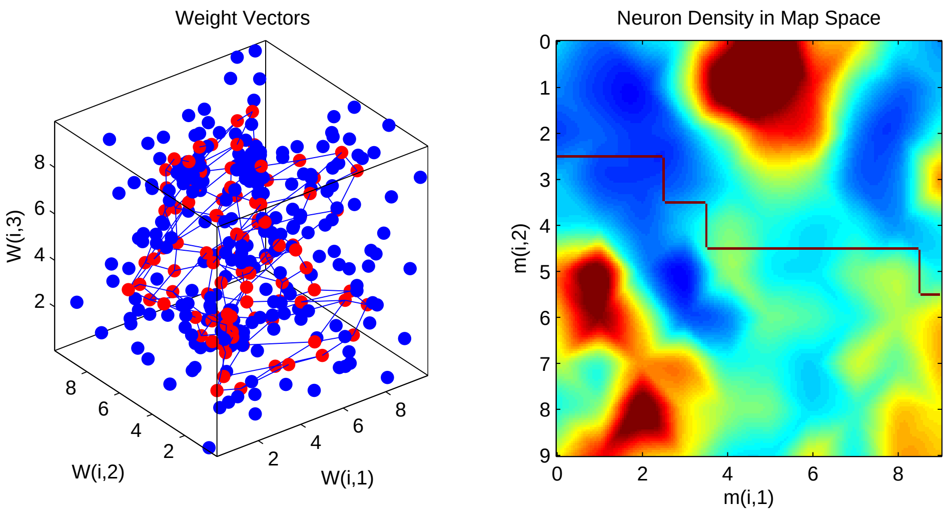

Figure 5.8: Weight vectors and neuron densities for adjacent adversaries under the random noise attack

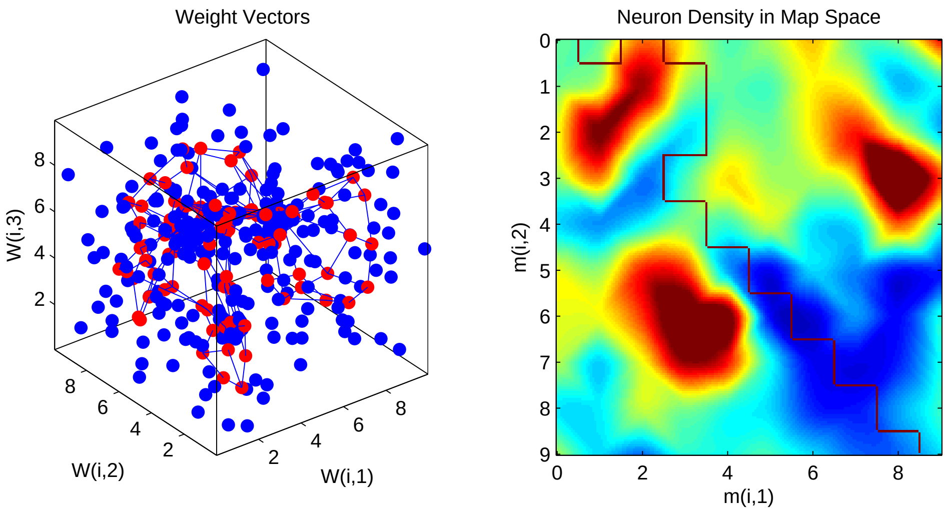

Figure 5.9: Weight vectors and neuron densities for equilateral adversaries under the random noise attack

Table 5.7: Error types and rates for different attack types using hierarchical clustering with equilateral adversaries

Error Type	None	Connect	Cluster	Noise
Pri→Sec	0	0	0	0
Sec→Pri	0.30	0.16	0.85	0.40
Adv→Pri	0.42	0.75	0.98	0.55

of chaff points the clustering algorithms can be forced to draw inaccurate decision boundaries and thus increasing the probability of error. Table 5.3 demonstrates that the strength of the attack increases regardless of its type as the number of chaff signals is increased. In Table 5.3 the numbers 0, 200, 400 and 600 correspond to no chaff points, one-third chaff points, two-third chaff points and full chaff points, respectively. These are derived according to number of signal points with which network is trained i.e. 600 signal points are used to train the network and then chaff points are added accordingly. Observe that the most strong attack comes out to be the point cluster attack, as it has high probability of error of classifying adversary user to be the primary user as compared to connection and random noise attack.

5.4 Real Signals: An Example

This section demonstrates the efficacy of the attack on features generated from real signal data. The scenario is that secondary users wish to distinguish the

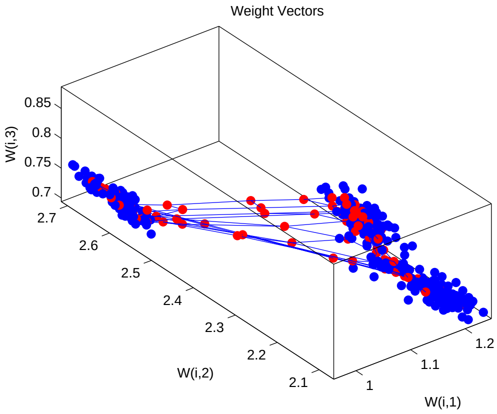

Figure 5.10: Weight vector (neuron) map for BPSK, 16QAM and FM signals

analog primary signal from the digital secondary signals. It is assumed the primary user is a frequency-modulated analog signal, secondary users are binary phase-shift keying (BPSK) and adversarial users are 16-point quadrature amplitude modulation (16QAM). While many different types of feature extractors are possible, this work selected three that were simple to compute and could be implemented as simple finite impulse response (FIR) filters. The first feature is the standard deviation of the time-domain signal itself. The second feature is the standard deviation of the time-averaged time domain signal, averaged over 10 samples. The third feature is the standard deviation of the derivative of the signal. In the end standard deviation of signals filtered with taps [1], [0.1, 0,1, ... , 0.1], and [1,-1] is computed. For the simulation, this thesis included small, random sampling and carrier frequency offsets of the input signals, consistent with an oscillator stability of 20ppm.

As a point of comparison, the USRP and USRPv2 respectively have oscillator

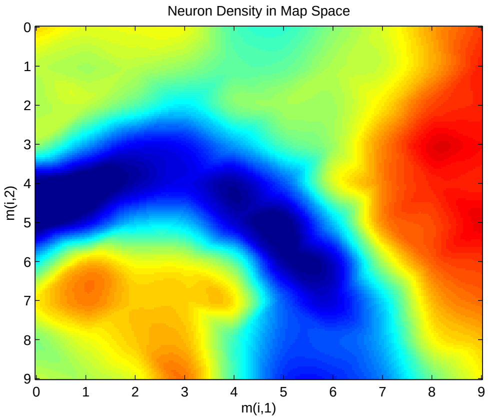

Figure 5.11: Density map for BPSK and 16QAM signals (notice the clear decision boundary between primary and secondary users)

stabilities of 50ppm and 2ppm [91]. It was also assumed the signal SNR has been normalized to approximately 10 dB, and included a scaling variance of 0.1. These random fluctuations in the input signal increase the variance of the input signal feature distributions in weight space. Figure 5.10 shows the self-organizing map that was trained to our signal inputs. The three classes are discernible, though the BPSK and 16QAM classes are close in weight space. In the associated density plot, boundaries are present between all three classes, but the strongest boundary is between the FM signals and the BPSK/16QAM signals. This result is an extension of [31] and demonstrates it is feasible to use neural networks with unsupervised learning and simple features to classify signals. The next experiment, demonstrates that the attacker can create input signals to manipulate the neuron structure. By fabricating signals with the appropriate feature vectors, the attacker can cause the 16QAM signals to

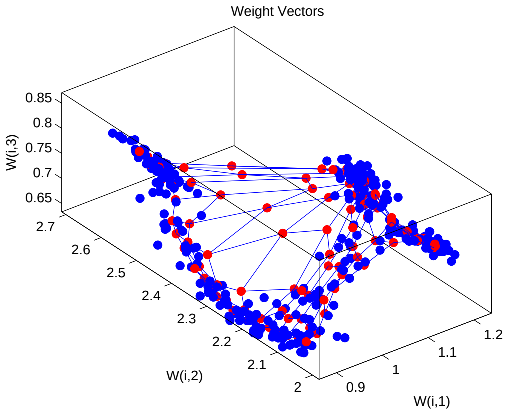

Figure 5.12: PSK signal is now a separate class, discernible from FM/QAM class

be lumped in with the FM signals. Given that with the exception of the computation of standard deviation, the feature transforms are linear, so the following rough approximation is used.

$$F(x_1(t) + x_2(t)) \approx F(x_1(t)) + F(x_2(t)) \tag{5.7}$$

Therefore, if in order to create chaff points to connect two signal classes, one simply needs to take random linear combinations of their time-domain signals. In particular if $x_{FM}(t)$ is the time-domain representation of our primary signal, and $x_{QAM}(t)$ is the time-domain representation of our adversarial secondary users, then one can create many signals that will generate the appropriate chaff by computing the following for random $\delta \in \text{unif}(0,\, 1)$.

$$x_{chaff}(t) = \delta x_{FM}(t) + (1 - \delta)x_{QAM}(t) \tag{5.8}$$

Figure 5.12 shows this attack. Note that due to the non-linearity of the standard deviation computation, the chaff points do not linearly connect the FM

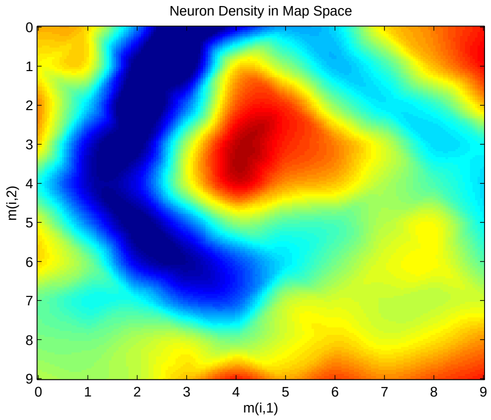

Figure 5.13: Chaff signals connecting two different signal classes i-e FM and 16-QAM signals, making them as one separate class in the density map

and 16-QAM clusters, but they achieve their end goal, none the less. In fact, the round-about nature of the points demonstrates the ability to create trails of chaff points that go around other clusters in feature space. With this attack, this thesis demonstrated the ability to manipulate decision boundaries and cause clusters of signals to be misclassified in a deterministic way. This results in a primary user emulation attack where the secondary user need not mimic the primary users signal.

Chapter 6

Mitigation and Conclusion

6.1 Mitigation

In the simulations above, there were always two classes i.e. the classification algorithm considered primary to be primary and all others to be secondary and grouped them in one class. In reality there are more than two classes as there can be more than one secondary user. That means each user will have its own cluster in the feature space. In this situation, a single boundary separating primary and secondary users is not helpful, in fact, it is problematic and confusing for the signal classifiers. So, if the estimate of number of classes is accurate the attacks against the signal classifier can be avoided, as in this way knowledge about number of classes would be available and a malicious user won't be able to fool the signal classifier. X-means algorithm [92] can be used to estimate the number of classes, or users, present. This algorithm would be more robust against signal classifier attacks as compared to K means itself [93].

The attacks against signal classifiers can also be thwarted if the secondary users know their own signal parameters and primary's signal parameters. Now, if the primary and secondary classes are known, the secondary users can use this information to perform a sanity check of the output of the unsupervised clustering algorithm. For example, it could detect primary and secondary user classes being collapsed down via a point cluster attack, and modify the approach used in performing the clustering. In many cases, this could simply be re-execution of the clustering algorithm. This approach could further decrease the misclassification rates in case of point cluster and connection attacks [93].

6.2 Conclusion

The use of signal classifiers and clustering algorithms for DSA has opened new frontiers of research in this area. There is a strong need to explore other types of signal classifiers and decision boundary algorithms which are less prone to attacks, more robust and efficient for DSA.

This thesis explored three types of attacks against signal classifiers. Self-organizing map (SOM) was used as signal classifier and K-means and weighted hierarchical clustering algorithm as decision boundary algorithms. First, it was shown analytically that this kind of attacks are possible for a simpler (one-dimensional) case. Then, more complex cases were simulated in Matlab. Simulations demonstrated the effectiveness of connection, point cluster, and random noise attack against signal classifiers. This demonstrated the fact that machine learning environment can be easy to manipulate by an adversary user, since if it can learn by the environment it can also be taught by the environment. So, a PUE attack can be launched very easily in this case.

Bibliography

[1] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Computer Networks: The International Journal of Computer and Telecommunication Networking*, vol. 50, pp. 2127–2159, 2006.

[2] H. E. Arslan, *Cognitive Radio, Software Defined Radio, and Adaptive Wireless Systems*. Springer, 1st ed., 2007.

[3] L. Berlemann and S. Mangold, *Cognitive Radio and Dynamic Spectrum Access*. Wiley, 1st ed., 2008.

[4] FCC, "Spectrum policy task force report 2002." `http://www.fcc.gov/sptf/files/SEWGFinalReport\_1.pdf`.

[5] FCC, "National broadband plan, chapter 5: Spectrum." `http://www.broadband.gov/plan/5-spectrum/`.

[6] DARPA, "The next generation program." `http://www.darpa.mil/sto/smallunitops/xg.html`.

[7] A. Fehske, J. Gaeddert, and J. Reed, "A new approach to signal classification using signal correlation and neural networks," in *IEEE New Frontiers in Dynamic Spectrum Access Networks (DySPAN'05)*, November 2005.

[8] T. Clancy, J. Hecker, E. Stuntebeck, and T. O'Shea, "Applications of machine learning to cognitive radio networks," *IEEE Wireless Communications*, vol. 14, pp. 47–52, August 2007.

[9] J. Mitola, *Software Radios: Wireless Architecture for the 21st Century*. Wiley, 2000.

[10] I. Mitola, J., "Software radios: Survey, critical evaluation and future directions," *IEEE Aerospace and Electronic Systems Magazine*, vol. 8, pp. 25 –36, Apr. 1993.

[11] R. Lackey and D. Upmal, "Speakeasy: The military software radio," *IEEE Communications Magazine*, vol. 33, pp. 56 –61, May. 1995.

[12] B. Tarver, E. Christensen, A. Miller, and E. Wing, "Digital Modular Radio (DMR) as a maritime/fixed Joint Tactical Radio System (JTRS)," in *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, vol. 1, pp. 163 – 167 vol.1, 2001.

[13] J. H. Reed, *Software Radio: A Modern Approach to Radio Engineering*. Prentice Hall, 2002.

[14] J. Melby, "JTRS and the evolution toward software-defined radio," *Proceedings Of IEEE MILCOM 2002.*, vol. 2, pp. 1286 – 1290 vol.2, Oct. 2002.

[15] FCC, "Notice of proposed rule making and order: Facilitating opportunities for flexible, efficient, and reliable spectrum use employing cognitive radio technologies, et docket no. 03-108, feb. 2005." `http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-03-322A1.pdf`.

[16] IEEE 802.11, "IEEE 802.21 working group on wireless local area networks." `http://www.ieee802.org/11/`.

[17] G. A. Fink, *Markov Models for Pattern Recognition.* Springer, 2008.

[18] R. A. Dunne, *A Statistical Approach to Neural Networks for Pattern Recognition.* Springer, 2008.

[19] K. Sankar and P. Wang, *Genetic algorithms for pattern recognition.* CRC-Press, 1996.

[20] E. Blossom, "GNU radio: Tools for exploring the radio frequency spectrum," *Linux Journal*, vol. 204, June 2004.

[21] M. Ettus, "Universal software radio peripheral (USRP)." `http://www.ettus.com`.

[22] M. McHenry, E. Livsics, T. Nguyen, and N. Majumdar, "XG Dynamic Spectrum Access field test results," *IEEE Communications Magazine*, vol. 45, pp. 51 –57, jun. 2007.

[23] R. Thomas, L. DaSilva, and A. MacKenzie, "Cognitive networks," *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 352 –360, Nov. 2005.

[24] F. Camastra and A. Vinciarelli, *Machine Learning for Audio, Image and Video Analysis.* Wiley, 2006.

[25] R. Herbrich, *Learning Kernel Classifiers.* MIT Press, 2003.

[26] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction.* MIT Press, 1998.

[27] T. Kohonen, "The self-organizing map," *Neurocomputing*, vol. 21, pp. 1–6, November 1998.

[28] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.

[29] K. I. Kim, K. Jung, S. H. Park, and H. J. Kim, "Support vector machines for texture classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1542 – 1550, Nov. 2002.

[30] N. Kim, N. Kehtarnavaz, M. Yeary, and S. Thornton, "DSP-based hierarchical neural network modulation signal classification," *IEEE Transactions on Neural Networks*, vol. 14, pp. 1065 – 1071, Sep. 2003.

[31] B. Le, T. Rondeau, D. Maldonado, and C. Bostian, "Modulation identification using neural networks for cognitive radios," in *SDR Forum Technical Conference (SDR'05)*, November 2005.

[32] M. Li, Y. Xu, and J. Hu, "A q-learning based sensing task selection scheme for cognitive radio networks," *International Conference on Wireless Communications Signal Processing (WCSP), 2009.*, pp. 1 –5, Nov. 2009.

[33] Y. Reddy, "Detecting primary signals for efficient utilization of spectrum using q-learning," *Information Technology: New Generations, 2008. ITNG 2008. Fifth International Conference on*, pp. 360 –365, apr. 2008.

[34] T. Clancy and A. Khawar, “Security threats to signal classifiers using self-organizing maps,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (Crowncom’09)*, June 2009.

[35] A. Khawar and T. Clancy, “Signal classifiers using self-organizing maps: Performance and robustness,” in *SDR Forum Technical Conference*, Dec. 2009.

[36] J. Hartigan and M. Wong, “A K-means clustering algorithm,” *Applied Statistics*, vol. 28, pp. 100–108, 1979.

[37] Q. Zhao and B. Sadler, “A survey of dynamic spectrum access,” *IEEE Signal Processing Magazine*, vol. 24, pp. 79 –89, May 2007.

[38] D. Hatfield and P. Weiser, “Property rights in spectrum: taking the next step,” *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 43 –55, nov. 2005.

[39] L. Xu, R. Tonjes, T. Paila, W. Hansmann, M. Frank, and M. Albrecht, “DRiVE-ing to the internet: Dynamic radio for IP services in vehicular environments,” *Proc. 25th Annual IEEE Conference on Local Computer Networks, 2000.*, pp. 281 –289, 2000.

[40] W. Lehr and J. Crowcroft, “Managing shared access to a spectrum commons,” *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 420 –444, Nov. 2005.

[41] C. Raman, R. Yates, and N. Mandayam, “Scheduling variable rate links via a spectrum server,” in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2005.*, pp. 110 –118, Nov. 2005.

[42] R. Etkin, A. Parekh, and D. Tse, “Spectrum sharing for unlicensed bands,” *IEEE Journal on Selected Areas in Communications*, vol. 25, pp. 517 –528, Apr. 2007.

[43] I. Mitola, J., “Cognitive radio for flexible mobile multimedia communications,” *IEEE International Workshop on Mobile Multimedia Communications, 1999.*, pp. 3 –10, 1999.

[44] M. Gandetto and C. Regazzoni, “Spectrum sensing: A distributed approach for cognitive terminals,” *IEEE Journal on Selected Areas in Communications*, vol. 25, pp. 546 –557, Apr. 2007.

[45] G. Vardoulias, J. Faroughi-Esfahani, G. Clemo, and R. Haines, “Blind radio access technology discovery and monitoring for software defined radio communication systems: problems and techniques,” *Second International Conference on 3G Mobile Communication Technologies, 2001.*, pp. 306 –310, 2001.

[46] N. Shankar, C. Cordeiro, and K. Challapali, “Spectrum agile radios: Utilization and sensing architectures,” *2005 First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 160 –169, Nov. 2005.

[47] P. Wang, L. Xiao, S. Zhou, and J. Wang, "Optimization of detection time for channel efficiency in cognitive radio systems," *IEEE Wireless Communications and Networking Conference, 2007.*, pp. 111 –115, Mar. 2007.

[48] Q. Zhao, S. Geirhofer, L. Tong, and B. Sadler, "Optimal dynamic spectrum access via periodic channel sensing," *IEEE Wireless Communications and Networking Conference, 2007.*, pp. 33 –37, Mar. 2007.

[49] Y. Hur, J. Park, W. Woo, K. Lim, C.-H. Lee, H. Kim, and J. Laskar, "A wideband analog multi-resolution spectrum sensing (MRSS) technique for cognitive radio (CR) systems," *Proceedings of IEEE International Symposium on Circuits and Systems, 2006.*, p. 4 pp., 2006.

[50] Y. Yuan, P. Bahl, R. Chandra, P. Chou, J. Ferrell, T. Moscibroda, S. Narlanka, and Y. Wu, "KNOWS: Cognitive radio networks over white spaces," *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007.*, pp. 416 –427, apr. 2007.

[51] G. Ganesan and Y. Li, "Agility improvement through cooperative diversity in cognitive radio," *IEEE Global Telecommunications Conference, 2005.*, vol. 5, pp. 5 pp. –2509, Dec. 2005.

[52] D. Cabric, S. Mishra, and R. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *IEEE Asilomar Conference on Signals, Systems, and Computers (Asilomar'04)*, pp. 772–776, November 2004.

[53] A. Ghasemi and E. S. Sousa, "Optimization of spectrum sensing for opportunistic spectrum access in cognitive radio networks," *4th IEEE Consumer Communications and Networking Conference, 2007.*, pp. 1022 – 1026, Jan. 2007.

[54] S. Jones, E. Jung, X. Liu, N. Merheb, and I.-J. Wang, "Characterization of spectrum activities in the U.S. public safety band for opportunistic spectrum access," *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007.*, pp. 137 –146, Apr. 2007.

[55] G. Ganesan and Y. Li, "Cooperative spectrum sensing in cognitive radio networks," *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 137 –143, Nov. 2005.

[56] D. Cabric, A. Tkachenko, and R. Brodersen, "Spectrum sensing measurements of pilot, energy, and collaborative detection," *IEEE Military Communications Conference, 2006.*, pp. 1 –7, Oct. 2006.

[57] T. Weiss, J. Hillenbrand, A. Krohn, and F. Jondral, "Efficient signaling of spectral resources in spectrum pooling system," *IEEE Symposium on Comm. and Veh. Technology*, Nov. 2003.

[58] D. Cabric, S. M. Mishra, D. Willkomm, R. Brodersen, and A. Wolisz, "A cognitive radio approach for usage of virtual unlicensed spectrum," *In Proc. of 14th IST Mobile Wireless Communications Summit*, June 2005.

[59] E. Visotsky, S. Kuffner, and R. Peterson, "On collaborative detection of TV transmissions in support of dynamic spectrum sharing," *First IEEE*

International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005., pp. 338 –345, Nov. 2005.

[60] J. Hillenbrand, T. Weiss, and F. Jondral, "Calculation of detection and false alarm probabilities in spectrum pooling systems," *IEEE Communications Letters*, vol. 9, pp. 349 – 351, Apr. 2005.

[61] M. Gandetto and C. Regazzoni, "Spectrum sensing: A distributed approach for cognitive terminals," *IEEE Journal on Selected Areas in Communications*, vol. 25, pp. 546 –557, apr. 2007.

[62] A. Ghasemi and E. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 131 –136, Nov. 2005.

[63] J. G. Proakis, *Digital Communications.* McGraw Hill, 5th ed., 2008.

[64] R. Tandra and A. Sahai, "Fundamental limits on detection in low SNR under noise uncertainty," *International Conference on Wireless Networks, Communications and Mobile Computing, 2005.*, vol. 1, pp. 464 – 469 vol.1, Jun. 2005.

[65] J. Ma, G. Li, and B. H. Juang, "Signal processing in cognitive radio," *Proceedings of the IEEE*, vol. 97, pp. 805 –823, may. 2009.

[66] A. Oppenheim, R. Schafer, and J. Buck, *Discrete-Time Signal Processing.* Prentice-Hall, 1999.

[67] W. Gardner, “Signal interception: A unifying theoretical framework for feature detection,” *IEEE Transactions on Communications*, vol. 36, pp. 897 –906, Aug. 1988.

[68] A. Dandawate and G. Giannakis, “Statistical tests for presence of cyclostationarity,” *IEEE Transactions on Signal Processing*, vol. 42, pp. 2355 –2369, sep. 1994.

[69] W. Gardner, “Exploitation of spectral redundancy in cyclostationary signals,” *IEEE Signal Processing Magazine*, vol. 8, pp. 14 –36, apr. 1991.

[70] B. Farhang-Boroujeny and R. Kempter, “Multicarrier communication techniques for spectrum sensing and communication in cognitive radios,” *IEEE Communications Magazine*, vol. 46, pp. 80 –85, apr. 2008.

[71] D. Thomson, “Spectrum estimation and harmonic analysis,” *Proceedings of the IEEE*, vol. 70, pp. 1055 – 1096, sep. 1982.

[72] W. Gardner, “Spectral correlation of modulated signals: Part 1–analog modulation,” *IEEE Transactions on Communications*, vol. 35, pp. 584 – 594, jun. 1987.

[73] W. Gardner, W. Brown, and C.-K. Chen, “Spectral correlation of modulated signals: Part 2–digital modulation,” *IEEE Transactions on Communications*, vol. 35, pp. 595 – 601, jun. 1987.

[74] H. Tang, “Some physical layer issues of wide-band cognitive radio systems,” *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005.*, pp. 151 –159, Nov. 2005.

[75] T. Yucek and H. Arslan, "Spectrum characterization for opportunistic cognitive radio systems," *IEEE Military Communications Conference, 2006.*, pp. 1 –6, Oct. 2006.

[76] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 201 – 220, Feb. 2005.

[77] Z. Tian and G. B. Giannakis, "A wavelet approach to wideband spectrum sensing for cognitive radios," *1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications, 2006.*, pp. 1 –5, jun. 2006.

[78] K. Challapali, S. Mangold, and Z. Zhong, "Spectrum agile radio: Detecting spectrum opportunities," *Proc. Intl. Symposium on Advanced Radio Technologies*, Mar. 2004.

[79] M. Oner and F. Jondral, "Cyclostationarity based air interface recognition for software radio systems," *IEEE Radio and Wireless Conference, 2004*, pp. 263 – 266, sep. 2004.

[80] T. Newman and T. Clancy, "Security threats to cognitive radio signal classifiers," in *Virginia Tech Wireless Personal Communications Symposium*, June 2009.

[81] A. Junkie, "SOM algorithm." `http://www.ai-junkie.com/ann/som/som2.html`.

[82] J. Burbank, "Security in cognitive radio networks: The required evolution in approaches to wireless network security," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (Crowncom'08)*, May 2008.

[83] Q. Mahmoud, *Cognitive Networks: Towards Self-Aware Networks*. Wiley, 2007.

[84] W. Stallings, *Cryptography and Network Security: Principles and Practice*. Prentice Hall, 5th ed., 2010.

[85] C. Stevenson, G. Chouinard, Z. Lei, W. Hu, S. Shellhammer, and W. Caldwell, "IEEE 802.22: The first cognitive radio wireless regional area network standard," *IEEE Communications Magazine*, vol. 47, pp. 130 –138, jan. 2009.

[86] A. Mody, R. Reddy, T. Kiernan, and T. Brown, "Security in cognitive radio networks: An example using the commercial IEEE 802.22 standard," *IEEE Military Communications Conference, 2009.*, pp. 1 –7, oct. 2009.

[87] R. Chen, J. Park, and J. Reed, "Defense against primary user emulation attacks in cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, pp. 25–37, January 2008.

[88] K. Bian and J.-M. J. Park, "Security vulnerabilities in IEEE 802.22," in *WICON '08: Proceedings of the 4th Annual International Conference on Wireless Internet*, (ICST, Brussels, Belgium, Belgium), pp. 1–9, ICST

(Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[89] Y. Zhang, G. Xu, and X. Geng, “Security threats in cognitive radio networks,” in *IEEE International Conference on High Performance Computing and Communications*, pp. 1036–1041, September 2008.

[90] T. Clancy and N. Goergen, “Security in cognitive radio networks: Threats and mitigation,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (Crowncom’08)*, May 2008.

[91] M. Ettus, “Building software radio systems.” `http://www.ettus.com/downloads/er_broch_trifold_v5b.pdf`.

[92] D. Pelleg and A. Moore, “X-means: Extending k-means with efficient estimation of the number of clusters,” in *Proceedings of the Seventeenth International Conference on Machine Learning*, (San Francisco), pp. 727–734, Morgan Kaufmann, 2000.

[93] T. Clancy, A. Khawar, and T. Newman, “Robust signal classification using unsupervised learning,” in *IEEE Transactions on Wireless Communications*, Under Submission.