Sustaining the Information Age: Channel Selection using Low-Computation Occupancy Analysis for Spectrum Sharing by Wireless Communication Devices

By

Quang Thai

A thesis submitted to Macquarie University

for the degree of

Doctor of Philosophy

Department of Engineering

September 2014

Typeset in LaTeX 2$_\varepsilon$.

Except where acknowledged in the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part for a degree in any university.

Quang Thai

Acknowledgements

This doctoral thesis is the culmination of several years of effort and patience. From the time I commenced my candidature to this point, I can say that I have experienced a gamut of emotions and moods. To say that the experience has been challenging would be an understatement. There are people who have given me the support, encouragement and warmth to pick me up and push me onward. This support has come in many forms, but all involved have made this undertaking a little easier. I would like to say a heartfelt 'thank you' to you all – there are too many of you to name exhaustively.

Firstly, I would like to thank my supervisor, Associate Professor Sam Reisenfeld, who encouraged me to embark on this doctoral endeavor in the first place. Despite spending seven years far removed from the world of academia, you still believed that I had the potential to succeed. I continue to be humbled by your wisdom and the sharpness of your mind. I will be most pleased if I can even partially emulate these attributes decades from now!

It would be remiss of me to not acknowledge the Department of Engineering at Macquarie University, whose staff have had faith in me from the beginning. They have provided me with the gift of opportunity – to complete this thesis, to teach and to travel to international destinations and mingle with researchers from around the world. I would like to say a thank you to Ana Borba and Emma Uy, who have been our indefatigable Department Administrators during my candidature. Your cheerful demeanour in spite of having to endlessly attend to my every whim and request should not go unmentioned. I would also like to thank Keith Imrie, who I am now convinced

is the world's most efficient and thorough proofreader. This thesis is immeasurably better due to your invaluable comments and feedback, and thanks to you I have a rediscovered appreciation of the nuances of the English language.

This thesis relied heavily on the corpus of spectrum occupancy measurements which has been made publicly available to the research community by RWTH Aachen University. I am thankful to Aachen University for doing so, and for providing me with access to the corpus.

Special mention also goes to my colleagues and fellow doctoral candidates at Macquarie University – who I now also count amongst my friends. I will not mention you individually as I risk inadvertently leaving someone out, but you all know who you are, and you have been a welcoming and reassuring presence in my university life.

Finally, I would like to thank my wife Linda Lowe – you have provided me with joy, support and comfort away from my research desk. The role you have played in my life during this journey is something I will always cherish.

Quang

List of Publications

- Q. Thai, S. Reisenfeld, K. Sithamparanathan, G. M. Maggio, *Energy-efficient spectrum sensing using cyclostationarity*, Proc. 2011 IEEE 73rd Vehicular Technology Conference (VTC Spring) (2011)

- Q. Thai and S. Reisenfeld, *Adaptive exploitation of cyclostationarity features for detecting modulated signals*, Proc. 2011 International Conference on Advanced Technologies for Communications (ATC), pp. 127-130 (2011)

- Q. Thai and S. Reisenfeld, *Deteriorative effects on feature-based signal detection due to imperfect training*, Proc. 2012 IEEE Wireless Communications and Networking Conference (WCNC), pp. 914-919 (2012)

- Q. Thai and S. Reisenfeld, *Energy detection techniques for signals subject to arbitrary channelizations*, Proc. 2012 IEEE International Symposium on Communications and Information Technologies (ISCIT), pp. 274-279 (2012)

Abstract

Recent years have seen a momentum shift emerging in the development of wireless communication devices, with software-defined radios advocated for devices which are flexible, adaptive and reconfigurable. The burgeoning demand for communication and data services 'anytime, anywhere' is placing unprecedented pressure on how radio frequency spectrum is provisioned and managed. The potential is there for cognitive radios to be able to identify channels which are underutilized and to exploit them. By effectively sharing radio frequency channels amongst several wireless networks, an increase in the communication capacity of the existing spectrum is made possible to meet future demand. In this thesis, techniques for supporting the requisite spectrum sharing are advanced. Signal processing and learning techniques are described to better comprehend channel usage in any given radio environment. The occupancy of a channel is predicted with a lower decision error rate by learning the spectral features present in transmitted signals. It is shown that devices can statistically characterize channels based on their occupancy rate and the unpredictability of their occupancy pattern. When coupled with learning, this allows intelligent and informed channel selection that considers the long-term benefit of exploitation against the costs of channel switching and interfering with other network users. The need to strive for computational efficiency to reduce power consumption, crucial in mobile communication devices, is a recurring theme that is addressed throughout.

Contents

List of Figures

List of Tables

List of Algorithms

1

Introduction

Digital communication systems have been sustaining the information age since the nascent ubiquity of personal computing. Nevertheless, in the past couple of decades, they have transformed the world in which we live in ways remarkable even compared to the emergence of the digital revolution. The most recent frontier that has been conquered has been *mobility* – a result of the ability to support high-data-rate, broadband networks *wirelessly*. It has become no longer necessary to use a bulky computer tethered to a wired connection, at home or the workplace, to take advantage of such broadband networks. Along with increasing miniaturization and computerization of consumer devices, we now live in a world where it is possible – and the ability to do so almost expected – to access information over high-speed wireless networks virtually anywhere, and at any time.

Furthermore, at no time previously has the world been so data-rich and information-capable. The amount of information being generated, stored, transmitted and retrieved on a daily basis, by people as well as machines, is staggering. Miniaturized computers are now embedded into an array of 'smart' devices – in fact, the *smart-* prefix is now liberally applied to refer to any device that has computer-enabled capabilities which make it faster, more responsive, more convenient or more useful in some way. Often coupled with wireless capabilities, such 'smart' devices have unprecedented capacity to connect and co-operate and share information amongst themselves and with their users, leading to the promise of smarter systems.

Australian statistics are available which support these observations, indicating that:

- An increasing proportion of Australians who are Internet users are choosing to do so through mobile and wireless networks [1], [2], [3], [4], [5] - see FIGURE 1.1.
- The overall volume of data that is being downloaded over mobile and wireless networks is increasing year-on-year at a notable rate - particularly where mobile telephony handsets are concerned [3], [4], [6], [7] - see FIGURE 1.2.

The accelerating popularity of other mobile devices such as tablet computers may place even further pressure on existing mobile broadband networks, given that such devices feature larger screens and are therefore considered to be more palatable and friendly to use than smartphones for data-intensive applications [8]. If the total amount of data being disseminated over wireless networks is to continue to increase, total capacity must be improved; this must be supported by one or more of the following:

- An increase in the amount of exploitable spectrum being found
- An increase in the efficiency at which existing spectrum is used

The need to increase the available spectrum that can be used for wireless communication systems is not solely motivated by Internet and mobile traffic demand. A lack of available spectrum may hinder the future development of innovative systems such as autonomous vehicular networks and the use of unmanned aerial vehicles (UAVs) in civilian applications [9].

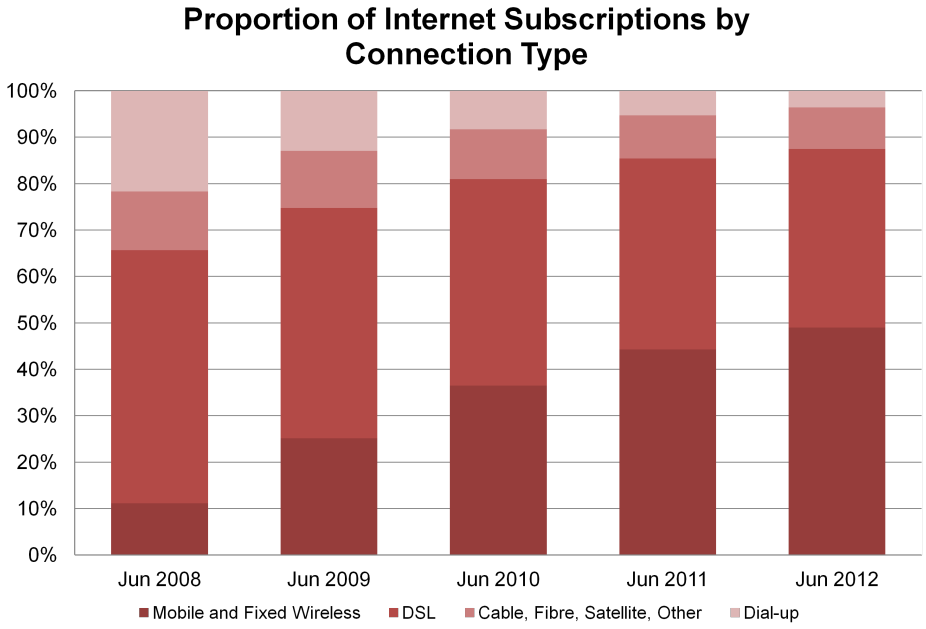

FIGURE 1.1: Over time, mobile and fixed wireless technologies have increased their overall share as the transport technology for Internet connection subscriptions, accounting for nearly the majority of such subscriptions in 2012.

Whilst there is active research seeking to solve this problem from both the aforementioned perspectives for supporting an increase in wireless capacity, this thesis is primarily concerned with the latter of these. In particular, some of the necessary ingredients for enabling *spectrum sharing* amongst different transmitters within a radio environment are considered.

Spectrum sharing is, however, retrograde to the traditional way in which radio spectrum has been managed. Access to spectrum has, for the most part, been according to a licensed access model, whereby exclusive rights to use allocated spectrum bands are paid for in licensing fees to a regulatory body such as the Australian Communications and Media Authority (ACMA) [10]. (There are exceptions to this, such as the ISM bands used for personal wireless networking technologies such as Wi-Fi and Bluetooth.) However, it has become apparent that this licensed access model has resulted in inefficient usage of spectrum - channels are often only used sporadically or for a small proportion of time, yet the licensed access model means that they cannot be exploited by any other devices. Spectrum sharing, in principle, is seen as a possible solution

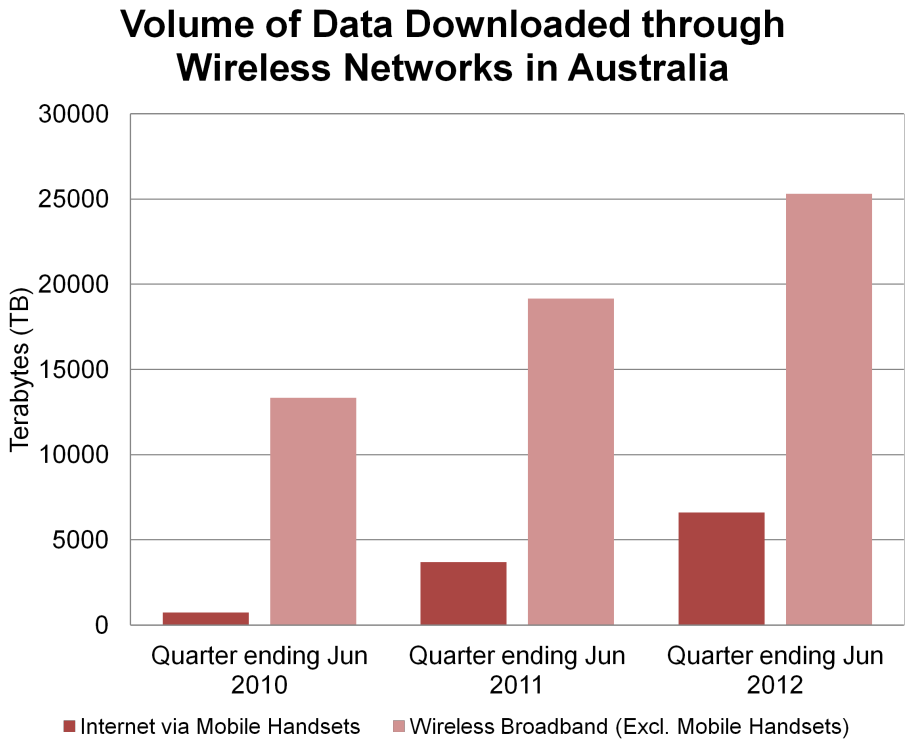

FIGURE 1.2: The volume of data downloaded through mobile telephony handsets and other wireless devices in Australia, during the quarter-year ending in June, over time.

to this inefficiency; however, there are regulatory and technical issues that need to be solved, with input from stakeholders such as licensed wireless users and those who seek to share spectrum with them, in order to make spectrum sharing a reality.

Whilst the legislative and regulatory concerns are important, this thesis is concerned primarily with the technical requirements and issues associated with spectrum sharing.

1.1 Evolving Towards Smarter Communication Devices – Some Necessary Ingredients

The progenitor of today's interest in dynamic, self-optimizing wireless communication devices can be traced back to when the concept of a *software-defined radio* was first established. From this point, the concept evolved to that of *cognitive radios* capable

of performing *dynamic spectrum access*.

1.1.1 Software-Defined Radio

Joseph Mitola III is credited with introducing the concept of software radios to the community [11], an idea that was presented as the next logical evolution in the way in which wireless devices are designed, engineered and manufactured.

Initially, wireless devices were built around fixed-purpose electronic circuits (hardware), and were therefore largely fixed-function. Over time, and noticeably within the past decade, such devices have gradually contained more software running on embedded microprocessors. Furthermore, increased competitive pressures on device manufacturers have meant that they are constantly interested in any approaches that can reduce the cost of getting a product from concept to market, and simultaneously reducing the cost of the engineering effort involved in this process. Device vendors are seeking any comparative advantage they can get in the 'technology race' [12]. These goals conflict with the lengthy timeframes involved in redesigning hardware circuits with each iteration or update of a product.

Software, on the other hand, is more suited to rapid iteration and shorter product cycles. Furthermore, a single hardware platform, consisting of a microprocessor or system-on-a-chip (SoC), can be embedded within multiple devices with different features or capabilities, simply by having different software running within them. This allows economies of scale to be exploited in the manufacturing and component acquisition process. In [11], Mitola saw software-defined radio (SDR) as a solution to promote interoperability of devices and communication systems in military applications and on the battlefield. He also saw his concept as a way of promoting rapid design of wireless radio devices through computer-aided design (CAD).

Initially constrained to the military research arena, the trend of increasing computerization of devices has has found its way into consumer products, made possible by the miniaturization of electronics and fast, power-efficient and cost-effective microprocessors [13], and mirroring Mitola's foresight in [11]. In conjunction with the increasing

speed and power-efficiency of microprocessors, and increasing computer memory densities, the software running in such devices has also progressively played an increasing role in 'defining' the device's capabilities - or, in the words of Mitola and Margulies, allowing a device to 'change personality' depending on the software loaded upon it [12]. From very simple code which simply switched in or out the hardware that was in use for a given device configuration, very complex software can now control all of the digital signal processing that takes place after the analogue RF antenna front-end and associated circuitry. The encroachment of software and digital technologies on the remaining functions traditionally implemented with analogue electronics will continue [13]. Modern microprocessor technology now brings platforms for software-defined radio [14], a concept first described over a decade ago, to the point of ubiquity.

The increasing role of software in wireless devices provides several potential benefits:

- **Flexibility**: Software provides a mechanism for implementing a multitude of communication protocols and standards, in a single device, in a compact way, rather than having multiple electronic circuits for each. A typical smartphone handset already can communicate via a plethora of wireless communication standards. This also addresses the *interoperability* concerns that Mitola had in military applications and networks.

- **Upgradeability**: Software can be updated far more easily than hardware in devices that operate in the field. This allows devices to be maintained more easily, and even be upgraded for new functionality, such as support for a new communication scheme or protocol. This also drives innovation and helps lower the barriers associated with getting new technology into the hands of consumers, as they could conceivably get new functionality through relatively simple product upgrades, rather than having to wait for an entirely new product to be available in the market before being able to do so.

- **Compatibility**: Closely related to *flexibility* and *upgradeability*, devices can have increased compatibility with multiple wireless networks. This also lowers the barriers to network operators in terms of rolling out new systems. By placing

in the hands of consumers devices that support multiple network technologies, operators can have both old and new networks co-existing, and *gradually* phase in the new technology, whilst devices can access either one depending on whichever one is available. This allows change to be made incrementally, avoiding the service disruption and costs that would be associated with a sudden blanket roll-out, and the consumer dissatisfaction associated with being compelled to buy a new device immediately [12]. An example of where this has been an issue has been the roll-out of mobile telephony networks implementing 2G, 3G and now 4G standards.

- **Dynamic behaviour**: Software facilitates devices that exhibit highly-dynamic behaviour during operation, with far fewer limitations than hardware-only implementations.

Figure 1.3 illustrates the concept of a software-defined radio described up to this point. The last of these points, *dynamic behaviour*, paves the way for yet another evolution in the software-defined radio concept – namely, *cognitive radio*.

1.1.2 Cognitive Radio

Later on, Mitola continued to develop his ideas, and along with colleagues at the Royal Institute of Technology, described his ideas on cognitive radios [15], made possible through the platform of software-defined radio [14] which could make them contextually and situationally aware, as well as having the ability to adapt to their users in very personalized ways. The main points here were that if devices and networks had complete knowledge about their own internal operational states, and the knowledge could be represented in terms of a common and formal language such as a *Radio Knowledge Representation Language*, then it is possible for devices to have meaningful 'conversations' with their network, allowing network resources to be planned and allocated more efficiently and provide an enhanced user experience. It is the flexibility and reconfigurability of SDR that makes it a foundation for cognitive radio.

Also described by Mitola was the *cognition cycle* – how a cognitive radio responds

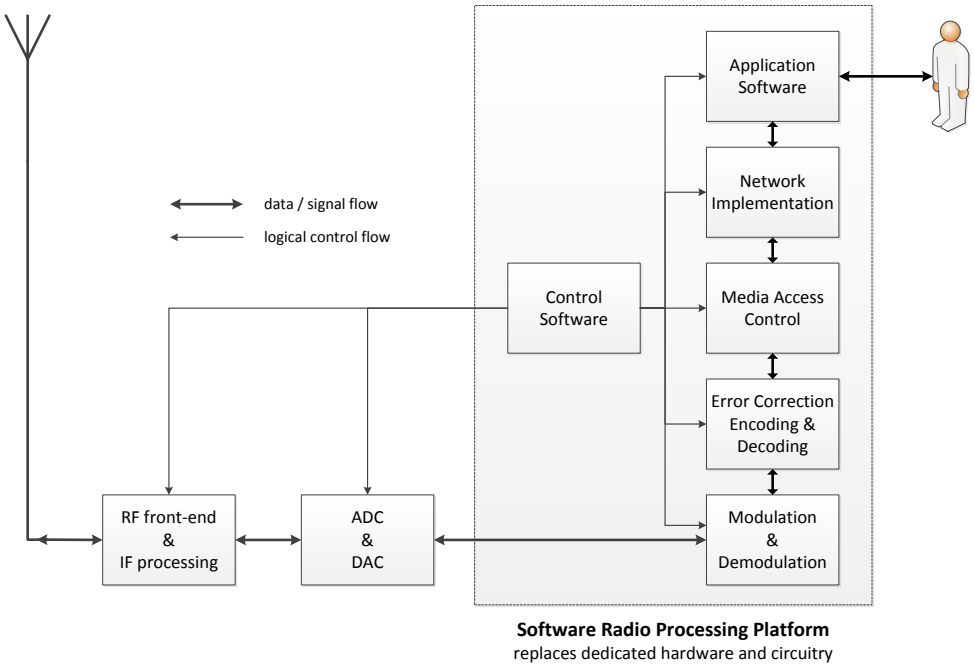

FIGURE 1.3: A software-defined radio architecture. A software radio processing platform runs software modules which replace network functions that would have been implemented in hardware in the past. This platform could be represented by DSPs, general-purpose processors, or some combination of these.

to interactions it has with its external environment [14]. The cognitive radio can infer its user's needs, or what the user might find desirable or useful, from external stimuli such as messages from the network, the user's voice, or other user input. It then considers the range of actions that it could take, prioritizes the actions it decides to take, allocates whatever computational and radio resources are needed to perform them, and then acts, thus exerting an influence back to the network and closing the loop. The relationship between actions taken and the observed stimuli fed back from the outside allows the cognitive radio to learn over time and improve the efficacy of its actions. Mitola's model indicates a *Learn* task but, rather curiously and perplexingly, it is illustrated purely as an information sink, and not an information source to other tasks in the cognition cycle. (Refer to Figure 6 in [14].)

In [16], Haykin also revisits the cognition cycle but, in his construct, inputs and outputs are purely in terms of the wireless network physical layer (i.e. stimuli consist

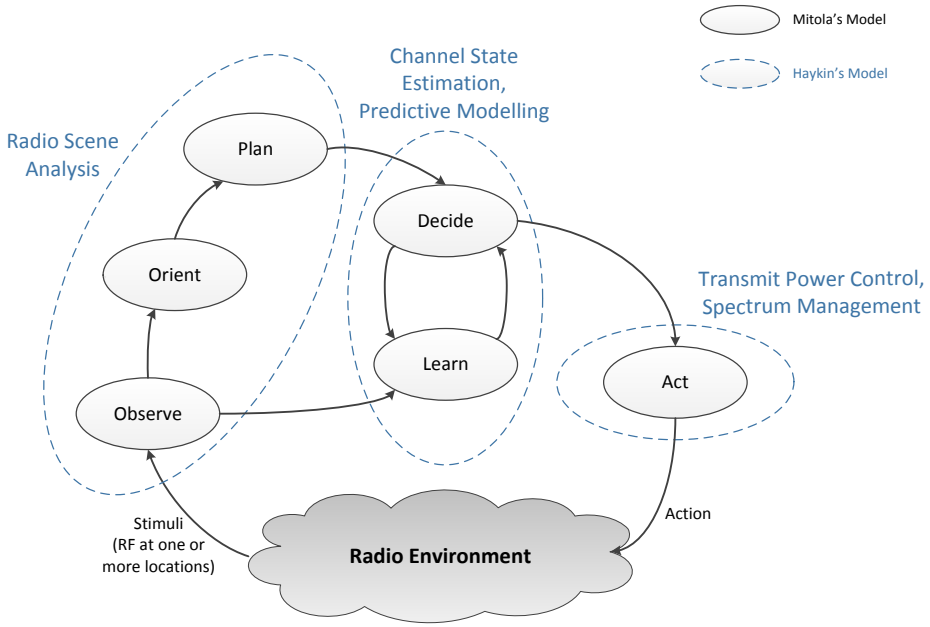

Figure 1.4: The cognitive cycle [14], [16]

of RF measurements, actions consist of transmitted signals, and the outside consists of the radio environment), whereas Mitola's formulation was deliberately more abstract and all-encompassing, and his examples wider-ranging in scope. However, Haykin proceeds to assign these tasks within the cognitive cycle to either the transmitter or receiver devices in a unidirectional communication link – although it is imaginable that other assignments are also, in principle, possible. (Bidirectional communication can be considered as two unidirectional links pointing in opposite directions. Both devices are *transceivers* and therefore must implement all tasks shown in the cognitive cycle.) Figure 1.4 illustrates elements of cognitive cycles described in [14] and [16], and an interpretation of how the tasks in the respective models correspond to each other.

Implementing the cognitive cycle on a SDR allows a wireless device to become dynamically self-aware, environmentally aware, able to learn from experience and self-optimizing. Rather than being fixed-function, its 'personality' can more easily be changed through the flexibility of software. These newfound capabilities provide wireless devices with the capability to address, as a collective, the issue of making more

efficient use of the available spectrum as described earlier in this Chapter. Haykin's definition of cognitive radio directly addresses this concern [16] – by being able to change operational parameters associated with the signal being transmitted by a wireless device, a cognitive radio seeks to provide a reliable communication link 'at any time, anywhere', as well as use radio spectrum more efficiently.

Wireless networks and systems are already designed to operate efficiently in their own right, according to the communication standards that they implement and adhere to in operation. Some flexibility in terms of mode of operation, such as choice of modulation, carrier frequency, channel, coding type, coding rate, transmission power, etc. are already possible, but these are limited by both the capabilities of the device and the communication standards involved. Cognitive radios in future devices may have even further scope for flexible operation and operating at higher efficiencies.

It is worth noting that 'efficiency' does not necessarily imply spectral efficiency (maximal effective data rate per unit spectrum); other objectives such as *power efficiency* (energy expended per effective bit of data transmitted) may be more important, depending on the scenario (e.g. a low-battery state). There are other desirable outcomes from wireless communication systems, such as maximal aggregate data rate, minimum latency, minimum outage probability, minimum interference on other devices, etc.. Multiple objectives can be combined into some form of overall *utility function* or *objective function* that measures a desired *quality of service* (QoS) – which the cognitive radio then seeks to tailor its operating parameters to maximize.

Consistent with the issue of perceived spectral scarcity described earlier, this thesis is primarily concerned with spectral efficiency *in the aggregate* – supporting a greater number of wireless devices than would otherwise be possible in a given radio environment through spectrum sharing principles. This requires cognitive radios to seek a new-found awareness of *spectral opportunities* within their radio environments, and act upon this knowledge, in ways that haven't been envisaged or deemed necessary in the past. The field of *dynamic spectrum access* hence came to prominence.

1.1.3 Dynamic Spectrum Access – A Capability of Cognitive Radios to Increase Spectrum Utilization

In traditional wireless networks, when multiple devices operating in a given network are in close proximity to each other, their signal transmissions tend to interfere with each other. Engineers have circumvented the interference problem, most often by forcing users to use different timeslots on a particular frequency band (TDMA), different subchannels within a frequency band (FDMA), different unique codes for either modulating the transmitted signal (e.g. DS-CDMA) or 'hopping' the signal between subchannels in a frequency band (e.g. FH-CDMA), or even some combination of these.

However, even in these cases, the allocation of timeslots, subchannels or codes to users is carried out according to a fairly static policy. This policy might be embedded within the devices themselves as a configuration parameter, or according to a scheme defined by a network operator and enforced at a *base station* with which all network devices register.

As self-optimizing devices that make decisions based upon an awareness of their radio environment, it is envisaged that cognitive radio devices will be able to avoid interfering with each other. Furthermore, by actively *detecting* which parts of the spectrum are 'vacant', and therefore exploitable by a device at any given time or place, wireless devices can use the spectrum much more efficiently, preventing the underutilization of spectrum that is often seen in spectrum occupancy measurement campaigns (e.g. [17], [18]). Thus, cognitive radios may realize the *spectrum sharing* concept that was described previously, embodying one of the solutions sought for raising wireless traffic capacity in wireless networks of the future.

Rather than simply utilizing whatever radio resources are allocated to them by the network to which they join, such as channel bandwidth or timeslot, cognitive radios performing dynamic spectrum access will be able to identify a range of *spectral opportunities*. The term *opportunistic spectrum access* is also used synonymously in some published literature (e.g. [19]). In terms of Haykin's cognitive cycle, this is part of the *radio scene analysis* task. Conceptually, cognitive radios are envisaged

as self-organizing agents that could allocate timeslots, subchannels or codes amongst themselves in a purely decentralized manner. In practice, however, they are likely to borrow elements from both this and the planned policies of today, thus occupying some space between these two extremes.

In [20], it is stated that the Federal Communications Commission (FCC) has identified three different spectrum sharing schemes which, together, will allow more wireless networks to co-exist in the available spectrum:

- **Unlicensed spectrum**: Any approved device is free to use this spectrum on a 'best effort' basis, without any attached guarantees associated with quality of service. A ubiquitous example of this is the use of WiFi wireless networking devices operating in the unlicensed Industrial, Scientific and Medical (ISM) band of frequencies at 2.4 GHz.

- **Underlay access**: Devices use a very low transmission power so as to avoid exerting excessive RF interference on licensed network users operating within the spectrum in question (by operating below their *noise floor*). To compensate for the low transmission power, a large bandwidth is used. Ultra-wideband (UWB) devices fall into this category.

- **Overlay access**: Devices co-exist with licensed network users within the spectrum in question, whilst using active techniques to avoid interfering with them.

Whilst devices using unlicensed access techniques such as WiFi have been a resounding success, the lack of guaranteed QoS and degraded service quality in densely-populated areas due to interference is an issue. UWB devices employing underlay techniques for short-range communications have failed to gain mainstream acceptance in the marketplace due to standardization problems [20], and the lack of success of the WiMedia Alliance. For some time there was significant research effort being directed at the *co-existence problem* between WiMedia-advocated UWB and WiMAX, as the two technologies had potentially overlapping spectral requirements when WiMAX was operating in the 3.5 GHz frequency band. Such research looked at ways in which UWB devices could have cognitive radio capabilities, dynamically detecting the presence of a

WiMAX signal and tailoring the spectral requirements of its UWB signal to 'avoid' interfering with it, thus allowing both a UWB and a WiMAX receiver in close proximity with each other to operate effectively (e.g. [21], [22]). However, the lack of penetration of both technologies in the market against competing technologies has resulted in a decline in attention in this area. Much research activity relating to dynamic spectrum access has been focused upon *overlay access*, and this continues strongly as various technical challenges still need to be overcome in this area.

Haykin himself presents a taxonomic description of spectral opportunities in [16] – therein, delineation is made based upon a three-stage 'quantization' of the amount of power present from other, potentially-interfering transmitters:

- **White spaces**: spectrum which is free from RF interference from other competing transmitters, but which nevertheless is subject to perturbations from ambient noise, thermal noise and other perturbations of a random nature.
- **Grey spaces**: spectrum which is occupied some of the time by competing transmitters; however, only a low level of signal power reaches the cognitive radio receiver from these transmitters.
- **Black spaces**: spectrum which is occupied some or all of the time by competing transmitters; a high level of signal power reaches the cognitive radio receiver from these transmitters.

Of these, white spaces are of most interest for dynamic spectrum access; grey spaces can be utilized but more care needs to be taken to avoid interfering with the existing transmitters operating within the spectrum of interest.

In [20], Roberson presents his own classification of the types of spectral opportunities that could be exploited by overlay techniques:

- **Unused**: There is not a licensed user of this spectrum even though it might have been allocated for a user who may appear at some time in the future (e.g. an unused television channel). This is the most beneficial and easiest type of opportunity to exploit.

- **Predictably used**: Licensed users of this spectrum transmit according to a temporal pattern that is well-defined in advance or very predictable (e.g. a television channel that only transmits at certain times of the day, beacon signals subject to predictable intervals).

- **Randomly used**: Licensed users of this spectrum transmit according to a random or unpredictable pattern, although some correlations of peak/trough usage may occur with certain times in the day [20] (e.g. mobile telephony bands). Cognitive radios performing dynamic spectrum access will need to be particularly careful to avoid interfering with such users.

- **Use with caution**: Licensed users of this spectrum may not do so regularly or often but, when they do, they *must* be given unimpeded access with high priority, as the users serve a critical, high-priority purpose (e.g. frequency bands used by emergency services).

Yet another taxonomy of opportunities for dynamic spectrum access is presented in [23]. The reasoning is that all wireless transmitters' signals can be measured by an occupancy or presence in each of the following *dimensions*, and that a cognitive radio can seek to co-exist with these by tailoring its transmission by operating orthogonally within these dimensions:

- **Frequency**: Not all channels within a frequency band of interest may be used. Unused frequencies represent an opportunity.

- **Time**: A transmitter is not constantly radiating an RF signal. Rather, it sends its frames in bursts, and there are times where the transmitter does not have any data to send and is therefore idle. Times when the transmitter is idle represent an opportunity.

- **Geographical Space**: The propagation loss in space means that a frequency or channel that is being used within one geographical area may be reused in another geographical area, so long as the areas are sufficiently separated in distance

such that the *mutual interference* falls below prescribed limits. This property is exploited for frequency reuse in cellular networks. Frequencies that satisfy this property with respect to transmitters that are sufficiently far away represent an opportunity.

- **Code**: A channel can be seen to support some number of transmitters using mutually-orthogonal codes adhering to some code design (spreading, time-hopping, frequency-hopping), even whilst they are transmitting at the same time and using the same frequency band. Unused codes on a channel represent an opportunity.

- **Radiation Angle**: A transmitter does not necessarily radiate its signal uniformly in all directions; there may be directionality associated with the transmission. If the receiver position is known, it is possible to avoid interfering with that receiver by either transmitting in a direction away from it, or modifying one's effective transmitter *radiation pattern* (e.g. through *beamforming* techniques) to generate a 'null' at that position, even if the same frequencies are used as that in the existing signal. Directions or angles that can be exploited in this way represent an opportunity.

According to [23], there is a relative abundance of research focusing on identifying opportunities along the *frequency*, *time* and *geographical space* dimensions, whereas there is a relative dearth relating to the dimensions of *code* and *radiation angle*. The authors maintain that a"*multi-dimensional spectrum awareness*" is needed for cognitive radios.

It is apparent that from the point of view of dynamic spectrum access, to borrow from a well-known cliché, not all spectral opportunities are created equal. Devices must take these distinctions into account in order to tailor their operation and make 'smarter' decisions in a truly cognitive fashion. Further chapters in this thesis will consider specific subsets of this problem more closely.

1.1.4 Performance Characteristics of Cognitive Radios Performing Dynamic Spectrum Access

Cognitive radios have many desirable characteristics but, just like any communication device or network, they are subject to the same engineering trade-offs that make it impossible to satisfy all of them simultaneously. Furthermore, the dynamic behaviour of such devices makes various aspects of their performance particularly difficult to assess or even quantify.

However, any device that reaches the market must ultimately be tested and assessed according to some performance metrics for certification and quality assurance purposes. Broadly speaking, such cognitive radios must:

- Be able to fulfill prescribed performance criteria relating to traditional network performance metrics (e.g. data rate, bit error rate, average latency)

- Be able to fulfill performance criteria relating to cognitive radios performing dynamic spectrum access (e.g. network establishment time)

- Avoid interfering with other wireless devices as much as possible, within defined limits

The following list provides an overview of what *quantifiable* performance metrics may be relevant for cognitive radios performing dynamic spectrum access – however, it is not intended to be an exhaustive list.

Detection Performance

In order for a wireless device to perform dynamic spectrum access, it needs to consider a frequency channel or band of interest, and make a reliable decision as to whether another user is currently transmitting on that channel or not. The *error rate* associated with this decision by a cognitive radio is a fundamental performance metric for dynamic spectrum access, as it relates to the ability to discriminate correctly between cases when there is another device transmitting within a certain bandwidth, and when there is not.

Objective of Dynamic Spectrum Access	**Role of Signal Detection**
Opportunistically identify *vacant* spectrum that it can use for its own traffic	Determine when a licensed user is not transmitting within a frequency band (licensed user is *absent*)
Relinquish currently-used spectrum that becomes *occupied* when a licensed network device starts transmitting on it, so as not to cause RF interference	Determine when a licensed user is transmitting within a frequency band (licensed user is *present*)

Table 1.1: The dual roles of signal detection in dynamic spectrum access

This task of determining channel *occupancies* by other wireless transmitters in this way is commonly referred to as *spectrum sensing*. The particular technique used to make the decision about such occupancy, within a defined bandwidth, is an instance of a *signal detection* algorithm, whose goal is reliable detection of the presence and absence of a licensed network user's transmitted signal. The duality of the above discrimination task results from two fundamental functions – both of which must be carried out by signal detection. These are summarized in Table 1.1. A cognitive radio does not want to transmit in such a way that it could interfere with licensed network users whilst, at the same time, it wants to be able to reliably detect when a vacancy in a frequency band does, in fact, exist so that it can make use of it.

These dual functions are often referred to collectively as *detect and avoid* [24].

According to Table 1.1, a frequency band can be considered to have two states: it is either 'vacant' and thus a cognitive radio can use it, or 'occupied' (not transmitting on it) and thus a cognitive radio cannot use it. Based on taking a *sample observation* of the signal in the frequency band of interest, the cognitive radio must, based on the evidence within the signal, make a decision as to which state the frequency band is in. As such, traditional signal detection is a *binary decision* problem, for which there are two kinds of *decision errors* which can be made:

- **False alarm**: A decision is made that the band or channel is occupied when it is, in fact, vacant. (This can also be referred to as *false detection* in some published works, such as [24].)

- **Missed detection**: A decision is made that the band or channel is vacant when it is, in fact, occupied.

It is desirable to ensure that both the *probability of false alarm* and the *probability of missed detection* are kept as low as possible in a cognitive radio system - however, in general, an engineering trade-off must be made between these two. This is considered in greater detail in section 2.1.1.

Network Join (Rendezvous) Time

According to [25], his is simply the time it takes for a cognitive radio node to:

- Detect the presence of another cognitive radio node (or nodes) that it wishes to establish a link with (e.g. using some form of unique signal 'signature')
- Declare to the relevant nodes that it wishes to establish a link (and perform any associated negotiation or handshaking)
- Be granted permission to transmit, according to channel parameters assigned by the network

Naturally, the lower this time is, the better.

Channel Abandonment (Evacuation) Time

In general terms, this is a measure of the amount of time it takes for a cognitive radio or, more accurately, all the cognitive radio nodes in a communications link, to 'abandon' or vacate a channel by ceasing their transmissions, in response to a licensed network user transmitting on that channel. The lower this time is, the less time is spent by the cognitive radio nodes exerting RF interference on the licensed users.

In [25], this time is quite specifically defined as *from the point at which the cognitive radio device detects a licensed network user transmitting on that channel.* This is further confirmed by the methodology that was used to measure the channel abandonment time for the XG cognitive radio nodes against DARPA-prescribed target values.

This definition is problematic, because it does not fully capture the true amount of interference that could be exerted. This is because *for a spectrum sensing schedule that involves taking periodic observations, a cognitive radio could unknowingly generate interference in between sensing events.* In other words, *interference could be exerted for a period of time significantly longer than the channel abandonment time.* In the network whose performance is evaluated in [25], it is apparent that this limitation applies to it, as the nodes only have a single modem chain for performing both communication and signal detection, and detection is performed using gaps in the transmission time.

As in [25], this is a metric that could be sensibly expressed as an upper bound value, albeit the definition should be modified to include the maximum time for which interference could be unwittingly exerted in between discrete sensing events.

Network Re-establishment Time

Once two or more cognitive radio nodes have decided to abandon a channel, they need to seek to re-establish a link using another vacant channel. This is the time it takes to carry this out – a lower value is clearly more desirable.

Whilst the XG nodes in [25] were provided a target of 500 ms by DARPA, it is plausible that a hard upper bound may be difficult to adhere to under all possible radio environments with varying occupancy patterns by licensed network users.

Temporal Sensing Overhead

Broadly speaking, this is a measure of the temporal *overhead* associated with performing spectrum sensing, applicable to cognitive radio systems whereby communication must be interrupted in order to perform spectrum sensing (e.g. because there is only a single antenna and modem chain). In other words, *it is the proportion of time where communication opportunity is forgone in order to perform spectrum sensing.*

For example, if spectrum sensing is performed every 5 s (the *sensing period*), and the signal detection task takes 0.5 s (depending on the literature, this may be called the *sensing time* (e.g. [26]) or the *detection time* (e.g. [27]), then the temporal sensing overhead is $\frac{0.5}{5} \times 100 = 10\%$.

The authors in [25] define a similar, related metric called the *white space fill factor* – this is basically the *complement* of the temporal sensing overhead, additionally scaled by the average *uptime* (expressed as a percentage) of the cognitive radio network, thus giving an overall proportion of time spent transmitting on a usable communications link.

In isolation, it is desirable to keep this value as low as possible. Mathematically, this implies having:

- The sensing time as short as possible
- The sensing period as long as possible

However, reducing the sensing overhead has implications elsewhere. For example, increasing the sensing period may adversely affect the channel abandonment time (see section 1.1.2). On the other hand, reducing the sensing time will, other things being equal, have an unfavourable effect on sensing accuracy – this is discussed in section 2.1.2.

Interference Power Level

Cognitive radios performing dynamic spectrum access will need to aim to minimize the RF interference that they exert on licensed network users.

They can do this using measures such as controlling their own transmit power to keep the interference seen by licensed users within acceptable bounds, and by vacating the channel that they are using in a timely fashion once the licensed users start transmitting on it (see section 1.1.2). The former is particularly difficult to do, as it requires a cognitive radio transmitter to have knowledge of the geographical locations of all licensed user devices, the geographical location of itself and its intended receiver with respect to the licensed users, as well as its own antenna radiation pattern.

The amount of interference can be measured at a licensed network device as simply the amount of RF power seen by its antenna that is attributable to the signal from a cognitive radio transmitter at another location. Care must be taken to distinguish

between the perturbation caused by the cognitive radio and that caused by ambient or thermal noise [25].

There are other metrics that are related to the interference level, in that they measure the amount of received signal perturbation seen by a licensed network device due to a cognitive radio transmitter, such as the *interference-to-noise ratio* (INR) [25] and the *signal-to-interference-plus-noise ratio* (SINR) [26]. An increased interference power level will cause an increase in INR and a reduction in SINR – this in turn degrades the performance of the licensed user's wireless link, by causing an increase in the *bit error rate* (BER) in much the same way as a lower signal-to-noise ratio (SNR) would in an interference-free scenario.

Power/Computational Efficiency

Fundamentally, this refers to the amount of additional power that a wireless device must consume in order to implement its cognitive radio capabilities, including algorithms for spectrum sensing/signal detection, machine-learning algorithms, powering the associated hardware and processors, and executing the associated software.

The *computational complexity* of cognitive radio software algorithms is often considered interchangeably with power efficiency – the less computationally complex the algorithms, the more power efficient the resulting implementation. This is because:

- Extra processor instructions in an SDR all consume energy
- A faster processor is needed to execute a more computationally complex algorithm than a less complex one in a given unit of time; processors which are clocked faster will generally consume more power

However, it is also generally the case that a more computationally complex algorithm will perform 'better' than a less complex one (e.g. a signal-detection algorithm), so an engineering trade-off frequently needs to be made between the two. Otherwise, it is desirable to keep cognitive radio implementations as computation- and power-efficient as possible – this has the following benefits:

- Battery drain is reduced for mobile devices.
- The manufacturing bill-of-materials for devices is reduced, as cheaper, slower processors can be used, helping keep costs down.
- Reduced power consumption reduces energy or battery recharging costs, and leaves a smaller 'environmental footprint'.

Adaptation Time

Broadly speaking, this refers to how quickly a cognitive radio 'responds' to a new environment, and dynamically adapts its behaviour accordingly in order to operate effectively and maintain a communication link. This may be highly dependent on the *rate of convergence* and training/learning time needed for the machine learning algorithms that it uses.

1.2 Challenges for Cognitive Radios

Cognitive radio is still largely a research concept, and continues to attract global research attention accordingly. There are various technical challenges that still need to be met in order for such devices to become ubiquitous and pivotal in coping with the spectral-scarcity problem. An overview is provided here – again, this list is not intended to be exhaustive.

1.2.1 Exposed Node and Hidden Node Problems

The signal detection problem, in its simplest form, involves taking a signal observation of a frequency band, making a decision as to whether there is another device transmitting on that band, and hence whether or not it is safe for the cognitive radio to transmit on it:

- If band is vacant – safe to transmit
- If band is occupied – not safe to transmit

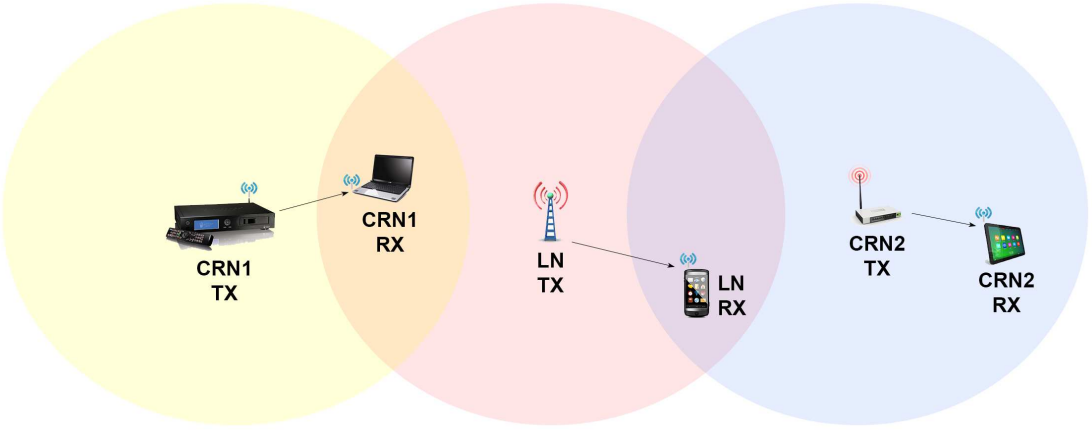

Figure 1.5: The exposed node problem is illustrated by the receiver performing signal detection in cognitive radio network 1 (CRN1) – the transmitter's radiation area is shown in yellow. The hidden node problem is illustrated by the receiver performing signal detection in CRN2 – the transmitter's radiation area is shown in blue. The 'licensed network' (LN) has a transmitter whose radiation area is shown in red, and whose signal is being sensed by CRN1 and CRN2.

However, this decision-making process can actually become overly simplistic when taking into account the spatial relationships between each of the transmitters and receivers involved in the radio environment. Figure 1.5 illustrates this point.

In Figure 1.5, there are three networks consisting of, for simplicity of visualization, one transmitter and one receiver each. There is a licensed network where the transmitter's signal covers the area indicated in red. There are two cognitive radio networks (CRNs) – the first one has a transmitter whose signal covers the yellow area, and the transmitter in the second one covers the blue area. Assume that, in both CRNs, it is the *receiver* that performs spectrum sensing, decides on whether a frequency band is vacant or occupied, selects an appropriate band to use, and tells its transmitter to use that band.

CRN1 illustrates the *exposed node problem*: CRN1 RX detects that LN TX is transmitting, as it lies within the red area. It tells CRN1 TX not to transmit on that band as it believes that doing so would interfere with the signal seen by LN RX. However, LN RX does *not* fall within the yellow area, so CRN1 TX could actually transmit to CRN1 RX, without exerting interference on LN RX, but is prevented from doing so by the signal-detection decision. This represents a *wasted spectral opportunity*

that is not exploited, due to CRN1 RX being 'exposed' to LN TX's signal.

On the other hand, CR2 illustrates the *hidden node problem*: CRN2 RX does not detect any signal from LN TX, because it lies *outside* of the red coverage area. It tells CRN2 TX that it may, therefore, transmit on that band. However, LN RX also falls within the blue area so, when CRN2 TX does transmit, it actually exerts interference upon it. This represents an interference scenario that is not detected, due to CRN2 RX being 'hidden' from LN TX's signal.

Avoiding these two problems is challenging, as it requires complete information about all the nodes' geographical positions as well as the transmitters' radiation patterns and directions – not just that of the cognitive radio nodes, but also the licensed network devices. Furthermore, signal detection is further perturbed by shadowing and fading effects due to geographical barriers and terrain, which add further uncertainty to not only the signal detection algorithm used, but also any measures that could be taken to mitigate these problems using measured signal strength [28]. Even if cognitive radios were equipped with positioning functionality as provided by the Global Positioning System (GPS), this adds a significant burden to the power budget of mobile devices. GPS also does not function effectively in all urban environments and, at any rate, it does not solve the problem of how to localize licensed users. (Retro-fitting all licensed users' devices with GPS functionality is not practical.) Mitigation of the hidden node problem is particularly important in order to convince licensed network operators that they will not be interfered with by a spectrum-management regime that allows for dynamic and opportunistic spectrum access by cognitive radios.

Techniques that attempt to deal with these problems include:

- Employing *co-operation* spectrum sensing between multiple, geographically diverse nodes that participate in the exchanging of their sensing results in order to attach a spatial 'picture' to, and reduce uncertainty associated with, the occupancy of a frequency band [29].

- Attempting to coarsely identify the location of licensed users through examination of individual spectrum sensing results of co-operating cognitive radios, and

enforcing a circular *protection region* or *disable region* around this location [30] – any cognitive radios within the protection region must not transmit on the same frequency band as that of the detected licensed user.

- Employing additional spectrum-sensing infrastructure that is spatially distributed (e.g. mobile-telephony base stations [31]), whose collective sensing results are combined in order to make a decision on what frequency bands are exploitable by a given cognitive radio device.

- Providing a centralized 'brokering service' [32] for assigning frequency bands to cognitive radios, which serves as a co-ordination authority that maintains a list of the frequency assignments to cognitive radios within a geographical area. Such services also incorporate knowledge of the geographical terrain, as well as requiring nodes to report their own transmit and receive antenna properties [32], such that propagation models can be applied and used to assist in determining the list of allowable frequency bands that can fulfill a request from a cognitive radio. A similar concept proposed in [33] is called a 'Radio Environment Map' (REM) server.

- Attempting to calculate a *hidden node margin* [34] – applied to detection *thresholds* used in cognitive radios when deciding if a frequency band is occupied or not, effectively placing an additional onus of 'proof' on the detector before declaring that it is vacant. The hidden node margin is calculated using statistical models based on the type of environment (e.g. open, urban, dense urban) and the relative heights of the licensed network users' antennas and those of the cognitive radio devices. It is based on the expected difference between the received signal strength on a frequency band between the two, whilst taking into consideration the statistical distribution of this value.

In general, problems that apply the decision-making process described above, without addressing the hidden/exposed node problems, are making the simplifying assumption that *all cognitive-radio-network and licensed-network nodes are sufficiently close to*

one another, and transmitted signals are sufficiently omni-directional, that it is valid to assume that any transmitted signal will cause sufficiently high interference on all nodes (although this is rarely, if at all, explicitly stated). In systems where a protection region is enforced, devices can operate without any change in the decision process (they are simply ordered to ignore a particular frequency band when located within the protection region). The same can be said for cognitive radio systems where a hidden node margin is enforced - this can simply be built into the threshold used by the signal detector to make its occupancy decision with respect to a frequency band. In both cases, since it is the hidden node problem that is being mitigated, the system errs on the side of offering additional protection to licensed networks, whilst incurring an opportunity cost in overlooking some possible transmission opportunities.

1.2.2 Prioritization of Cognitive-Radio Traffic

It is implicitly assumed in much of the spectrum sensing literature that when a cognitive radio performs spectrum sensing, it is actually detecting the signal associated with a licensed user. However, it is possible that the detected signal actually belongs to another cognitive radio transmitter that is opportunistically utilizing the frequency band of concern. The 'avoid' action associated with finding occupied spectrum effectively means that the first cognitive radio device that is able to claim it for its own use can effectively 'lock out' other cognitive radio devices from using it (until, at least, it is forced to relinquish it due to the licensed user occupying it, after which all cognitive radio devices are able to compete for it again). This is the so-called 'finders-keepers' scenario [28].

There is a certain unfairness associated with this scenario that may be unpalatable – to alleviate this, some form of arbitration or scheduling mechanism to provide fair-share access for all cognitive radios to a spectral opportunity is required. This could be taken further to prioritizing different cognitive radios and their traffic, and scheduling access accordingly. For example, in [35], a TDMA-based approach is used to divide vacant channels into time slots which are allocated to different cognitive radios and traffic types in military applications – both centralized assignments and distributed

assignments using the Unifying Slot Assignment Protocol (USAP) are considered.

1.2.3 The Costs of Spectrum Sensing, and Selecting Sensing Parameters

A cognitive radio performing dynamic spectrum access is required, in addition to transmitting/receiving data as part of its basic communication function, to perform spectrum sensing in order to perform its 'detect and avoid' function.

Implementing this function in a wireless device incurs various costs. The nature of these costs depends on how this functionality is implemented. In broad terms, spectrum sensing can happen in *discrete time* or *continuous time*.

In discrete-time sensing, this task takes place at certain time *epochs*, frequently periodically spaced (according to some pre-defined *sensing period* or *detection period*, in which case it can also be referred to as *periodic sensing*. In between these epochs, the device is free to carry out its usual communication tasks, using whatever spectral resources and frequency band(s) it has selected for its own use. However, there is an *opportunity cost* associated with the spectrum-sensing activity as, unless the device has two antennas and associated modems that can be operated independently, it can either perform communication *or* spectrum sensing, but not both simultaneously. On the other hand, the opportunity cost of sensing is counter-balanced by the 'cost' of (unknowingly) interfering with a licensed user for an extended period of time by leaving a long time interval between sensing epochs, as these epochs become the only points in time where the device can update its knowledge of the channel state and take action accordingly. This contention gives rise to questions such as: *what is the 'optimal' rate at which spectrum sensing events should be performed?*, or, *what is the minimum sensing rate that can afford the required level of protection of licensed network users?* Works such as that presented in [36], [37] and [38] address questions such as these. However, it is envisaged that the protection of licensed network users may be of such paramount importance in many environments that traditional optimization formulations may be of limited use as the cost of interfering with these users approaches infinity – such

scenarios are particularly challenging for discrete-time sensing regimes.

Continuous-time sensing, in theory, offers more protection for licensed users, because sensing takes place all the time, *in parallel* with communications. However, this requires two antennas and associated modems operating in tandem. Even then, there are technical hurdles associated with such an implementation. For example, there is the problem of preventing 'self-sensing' when a cognitive radio transmits on one antenna (for communication) whilst simultaneously performing spectrum sensing on another antenna in close proximity [21], requiring a perturbation-subtraction scheme not dissimilar to active noise cancellation techniques used in audio reproduction. Also, continuous sensing is more power-demanding.

In either case, the signal detection algorithm consumes computational resources and hence power, which has implications for mobile handsets, particularly for implementations where sensing is performed continuously or subject to short sensing periods. Besides selecting an appropriate sensing period, it must also select a *sensing time*, defined as the length of time spanned by a signal observation used by the detection algorithm to make its occupancy decision. It is generally known that there is a trade-off between the sensing time and sensing performance (in terms of accuracy): longer observation times, under certain assumptions, and with other sensing parameters being equal, result in fewer sensing errors – this is explained further in section 2.1.2. However, longer observations mean that more power is consumed, and there is a larger opportunity cost associated with forgone communication in discrete-time sensing. Researchers have attempted to formulate optimization problems with the aim of selecting a minimum sensing time that balances these competing requirements, whilst still meeting prescribed detection performance criteria (e.g. [27]). However, other research work considers the effect of long sensing times and challenges the assumption that this necessarily leads to better detection performance (e.g. [39]) – the authors point out that long sensing times mean that the licensed user may actually stop/start transmitting several times during the observation, adversely perturbing the signal detection algorithm. Real-life factors such as this further complicate the answer to this question.

Another real-life factor, *mobility*, complicates the selection of sensing parameters

such as sensing period and duration, whereby shadowing effects that can rapidly change may be a characteristic of the radio environment (e.g. [40]).

1.2.4 Characterization of Spectral Opportunities

As previously described in this chapter, spectral opportunities and temporal vacancies can be characterized in several different ways, with accordingly different 'payoffs' to a cognitive radio that attempts to exploit them. Therefore, there is a general question where *given a range of spectral or temporal opportunities that it could exploit, how does a cognitive radio pick the 'best' one at any given time?*

Broadly speaking, if the spectral opportunities can be considered as a range of frequency bands or channels, this is a problem of formulating a *channel selection policy*, which requires:

- Each channel or band to be characterized according to one or more traits that influence how much *reward* or *cost* is associated with using it.
- A channel-selection policy to be formulated, where one channel or band is selected preferentially over another based on their relative characterizations. The selection policy could be based on heuristics, pre-defined rules or *learned*.

In [41], channels are characterized based on the overall proportion of time that they are occupied by a licensed user (referred to as a *collision probability*), as well as the amount of noise (and hence experienced SNR) prevailing, and learning of a channel selection policy is implemented via a 'stochastic learning automaton'. Time-slotted access is assumed, and it is also assumed that the licensed users only transmit in short bursts equal in length to one time slot.

In [42], channels are characterized along two axes - their *duty cycle* (synonymous with collision probability in [41]), and their *complexity*, which is a measure of the 'randomness' or unpredictability of the occupancy patterns exhibited by the licensed user. The authors specifically use *Lempel-Ziv* (LZ) complexity (described in [43]) in their work. They point out that, even with channels that have the same duty cycle, they

can have different LZ complexities; higher complexity means less underlying regularity in the occupancy patterns, making it more difficult for learning algorithms to converge to a useful and consistent policy. Conceptually, the duty cycle influences the amount of transmission opportunity available to a cognitive radio, whereas the complexity measures the difficulty associated with exploiting the opportunity. The authors advocate the use of *Q-Learning* for model-free learning of channel selection policies, as well as asserting that *there is a higher probability of the Q-Learning algorithm failing to converge to a policy for a highly complex set of channels compared to a corresponding set of channels with low complexity.* However, the learning is applied to *homogeneous* sets of channels only, rather than a *heterogeneous* environment with many different channels exhibiting a wide gamut of duty cycles and complexities.

1.2.5 Rendezvous Between Cognitive Radio Devices

If it is assumed, for simplicity, that there is a unidirectional wireless link from a transmitter to a receiver in a cognitive radio network, then the following problem emerges:

- RF interference must be avoided at the *receiver* location, to ensure that the receiver can see a clean signal from the transmitter
- The receiver is therefore in a position where it can directly perform spectrum sensing to identify what channels are vacant or not
- However, it is the *transmitter* that must change the frequency band used accordingly, so that a vacant channel is used
- Therefore, the receiver must have some pre-allocated channel (sometimes referred to as an *order-wire* [15]), known to both the transmitter and receiver device, in order for the receiver to tell the transmitter what frequency band it should use, or when it should switch bands because the one presently being used is no longer vacant

In conventional wireless devices, the spectral resources that they need to 'find' each other is dictated by pre-assigned or licensed spectrum, or protocols relating to access of

unlicensed spectrum (e.g. that used in Wi-Fi) that are built into all complying devices. Everything that is needed to enable such devices to *rendezvous* is supplied and known in advance. Conceptually, it is envisaged that cognitive radios will self-organize and self-allocate these resources (in [25], this is the "no pre-assigned frequencies" requirement), but they can only negotiate these if they have some means of communicating their needs to each other in the first instance – the so-called 'chicken-and-egg' problem.

Use of a pre-assigned, common control channel that is used by all devices performing dynamic spectrum access in an area is a frequent assumption in the research literature, upon which devices can broadcast information about themselves and rendezvous other devices in the initial instance. However, there are issues that must be dealt with, such as provisioning the spectrum associated with the common control channel such that it can scale with the number of devices in the area and does not become saturated [40]. Such network architectures can be viewed as a hybrid between traditional and cognitive approaches – alternatively, as a 'supported' cognitive environment whereby some common support infrastructure is provided to cognitive radios for them to rendezvous and then operate autonomously. It is noteworthy that *the control channel must always be available, and protected from interference from other devices in the same way that licensed spectrum is today.*

On the other hand, the restrictiveness associated with the common control-channel-approach is eschewed in works such as [44]. Instead, they advocate cognitive radios inserting *cyclostationary signatures* into their transmissions, which other devices can scan their exploitable spectrum for – once the signature is found, the spectral location of the signature also indicates the frequency band being used, and provides a frequency estimation mechanism, allowing the transmitter and receiver device to 'join' each other on the network, achieving rendezvous. However, some assumptions are made – orthogonal frequency division-multiplexing (OFDM) modulation is assumed as it allows signatures to be easily created and made part of the modulation process, and the authors note that a mechanism is needed to assign signatures to cognitive radio networks (e.g. some regulatory or approval authority). Also, the signatures require that some of the subcarriers of the OFDM symbols being transmitted be allocated

to signature generation – thus, its generation represents a communication overhead. Finally, the issue of how a change in the frequency band used by the network is carried out once rendezvous has been achieved, in response to the band now becoming occupied, and without a protected control channel, is not clearly addressed.

1.2.6 Security and Privacy

As cognitive radio and dynamic spectrum access research has matured, attention has started to turn towards some of the system security and privacy concerns, the nature of exploits or attacks that cognitive radio networks may be subjected to, and how they might be obviated.

The following is a list of some of the concerns which have been identified – this list is not intended to be exhaustive:

- Dynamic spectrum access follows, conceptually, a *listen-before-talk* approach, whereby devices 'listen' to see if a frequency band is vacant before using it. Such an access control approach is vulnerable to a *denial of service* attack. For example, a malicious device could attempt to 'lock out' cognitive radios from accessing a spectral resource by transmitting in white and grey spaces. The cognitive radios will find that the frequency band is not vacant, and hence attempt to look elsewhere for a vacant band to use. This could be implemented by a cognitive radio trying to 'monopolize' its access to a particular band through the 'finders-keepers' scenario (see section 1.2.2). It can be considered a form of *jamming* [45].

- In a subtle variation relating to *co-operative spectrum sensing* architectures (where the signal detection decision is made by combining the individual decisions made by several different nodes), it is possible for a device to *falsify* its own local spectrum sensing report, in order to try and influence the decision that is ultimately made as to whether a frequency band is vacant or not, in order to attempt to obtain exclusive access to a channel [40].

- A further variation that can lead to denial of service is a *primary user emulation* attack [45], [46]. Here, a cognitive radio tailors its modulation so that its RF transmission mimics that of an expected *primary user* (i.e. licensed user), with the same result of denying the capacity to dynamic spectrum access by cognitive radios performing spectrum sensing.

In general, many cognitive radio architectures involve participating nodes collaborating and sharing large amounts of information with each other for collective gain, but there is a conceptual trade-off between this and privacy concerns, particularly in schemes that make use of the geographical location of nodes. The authors of [40] make note of this particularly for cognitive radio networks in vehicular applications, where the potential exists to eavesdrop and possibly track the location of drivers.

1.2.7 Suitability for Different Traffic Types

Dynamic spectrum access has been touted as a beneficial solution for various applications.

There has been much attention, for example, about the underutilized spectrum associated with unused television channels in different geographical areas, and how it can be exploited by cognitive radios for wireless data services (e.g. [47], [48]). The IEEE 802.22 standard specifically caters for devices using cognitive-radio techniques for exploiting white space in spectrum allocated for television broadcasting to provide data services over wireless regional area networks (WRANs). Accordingly, the FCC first expressed in-principle agreement for allowing access to licensed television white space for unlicensed use by cognitive radio devices in 2008 [49]; they continue to assess and refine the rules that will govern approval and certification of such devices. Regulatory authorities such as the ACMA have consulted the wider community about how to allocate and utilize the additional spectrum (the *digital dividend*) that will be freed as analogue television broadcasts are phased out in favour of digital television (DTV) [50] – a decision has been made to allocate the digital dividend, along with spectrum in the 2.5 GHz band, to interested parties seeking to provide advanced wireless communication

services [51], with the auction process scheduled to begin in 2013 [52]. The favourable propagation characteristics of the frequencies in the digital dividend will make it highly sought after and, at the same time, this makes it ideal for realizing wireless data networks.

On the other hand, cognitive radio technology has also been proposed for emergency-service networks at disaster zones (e.g. [53], [54]) – as such devices are able to adapt to their radio environment, determine and exploit all available spectral resources, maximize efficiency and capacity, overcome interoperability concerns that often plague traditional networks and devices, and self-organize into functioning wireless networks autonomously and with minimal setup and intervention from emergency-services crew in a crisis situation.

However, there are some question marks over what types of traffic cognitive radio networks are truly suitable for. Overlay cognitive networks suffer from difficulty in being able to guarantee a minimum QoS, due to the stochastic nature of the white space in the radio environment for them to exploit. It is difficult to imagine that this would be tolerable for an emergency services network – an application with extremely high-priority traffic. In [42], the authors express the view that cognitive radios exploiting grey spaces are best suited for delay-tolerant traffic, such as machine-to-machine (M2M) applications.

Some researchers have attempted to exploit both overlay and underlay cognitive radio techniques for demanding traffic applications, in order to exploit the favourable characteristics of both forms. In [55], the authors use this hybrid approach for H.264 video applications. Real-time video streaming over wireless networks is a challenging proposition due to the time-sensitive nature of the traffic, and the video quality degradation that results when the decoder does not receive data in a timely fashion. The principle here is that the video is logically encoded into a *base* layer and an *enhancements* layer, with the former transmitted using underlay techniques, and the latter using overlay. The latter is most affected by interruptions due to the frequency band being used becoming occupied by a licensed user; however, as this does not affect the base layer, even with only partial enhancements-layer information being presented to

the decoder, acceptable video quality can be achieved [55].

1.3 Contributions of this Thesis

Engineering concepts and solutions for wireless devices which can perform dynamic spectrum access continue to be developed. In this section, the contributions of this thesis in this area are identified, as well the manner in which they build upon, extend, or improve outcomes that have been achieved previously.

1.3.1 Exploiting the Presence of Distinguishing Signal Features for Efficient Detection

The importance of reliable signal detection algorithms in spectrum sensing by cognitive radios has already been elucidated. Various approaches have been considered for such applications (e.g. see [23]). Some techniques, such as *cyclostationarity* detection schemes, make use of additional *features* of a signal beyond its PSD.

Exploiting this additional information results in a more accurate and reliable signal detector, but is generally more computationally intensive and demanding of power. In [56], the detector makes its decision based on a test statistic involving a summation over the entire *cyclic spectrum* of a radio signal observation. This can be computationally wasteful, since the cyclic spectrum must be calculated in its entirety for each detection operation, including regions whereby there are no distinctive features in the signal being detected.

In Chapter 2, it is demonstrated that it is possible to dramatically reduce the amount of computation required of a cyclostationarity detector, by concentrating on the regions of the cyclic spectrum where the features specific to the transmitted signal being detected are 'strongest' and most evident. It is also demonstrated that, in the absence of prior knowledge about the locations of the requisite features on the cyclic spectrum plane, it is possible to use learning algorithms *to discover these features. This is an important capability, since different cyclostationarity features will manifest themselves depending on*

specific characteristics of the transmitter and the modulation schemes that it employs. It is demonstrated that the benefit of improved detection performance is still present, even though the amount of computation is reduced.

1.3.2 Self-Configuring Signal Detectors When Operating in the Field

In the previous section, it was mentioned that it was possible to have a cyclostationarity detector which can learn which regions of the cyclic spectrum to 'home-in on' to exploit in its detection decision, simultaneously reducing the computational complexity and power consumption whilst still providing superior detection performance. However, the learning that was demonstrated took place in contrived conditions. Learning was reliant upon having a *training set* of signals which were truly candidate representations of the transmission which must be detected. However, as an autonomous agent operating in an arbitrary radio environment, a cognitive radio has no fully reliable way to obtain such a training set, nor is one simply provided to it as a basis for learning.

The implication here is that it must obtain the training set from signal observations that it takes itself. However, if the transmitter in question does not transmit all of the time, and it is not possible to know in advance when it will transmit, then any signal observation taken may not be actually representative of the transmitter, and therefore its use during the training process leads to an inferior training outcome. However, this problem is actually impossible to avoid - as to do so would imply that a perfect signal detector already exists to detect with perfect accuracy when a transmission is present. It is only possible to devise a detection scheme that allows some engineering control over the extent to which the problem can be tolerated.

A two-stage signal detection scheme is described: the first stage is based on an *energy detector*, and its role is pivotal to the *training* phase, whilst the second stage is a low-complexity cyclostationarity detector, which is used during the *operational* phase. There exists published work describing two-stage detectors to achieve other objectives in relation to spectrum sensing. For example, [57] use a two-stage detector

for performing coarse followed by fine sensing when looking for a vacant channel to use across a large bandwidth, with the advantageous outcome being speed of locating a vacant channel. In [58], the authors use two energy detector stages, where the second stage provides a 'second opinion' when the detection outcome of the first stage appears inconclusive, resulting in improved detection performance compared to a single energy detector. However, this work specifically addresses the problems and trade-offs associated with '*imperfect training*' in a cognitive-radio spectrum-sensing context.

In Chapter 3, some of the implications of the above issues in terms of training time and detection performance are explored. By leveraging the desirable qualities of two different signal detection schemes, both the training and operational requirements of a cyclostationarity detector are addressed whilst operating in the field.

1.3.3 Identification of Spectral Opportunities Subject to Arbitrary Channelizations

Traditional signal detection schemes declare frequency bands or channels as being vacant or occupied (by a licensed network user). This is the only decision that is made; it is assumed that the *spectral boundaries* (and hence the *centre frequency*) of the channel has been defined elsewhere in advance.

An assumption that is often made is that, when performing spectrum sensing, the bandwidth of interest is subdivided into a certain number of equally-spaced, equal-width frequency bands (e.g. [57]). This makes sense for some present applications whereby a cognitive radio may seek to exploit vacant channels in licensed networks which make use of well-defined channels, located within spectrum whose usage is tightly controlled by regulatory bodies. However, if an increasing number of wireless devices with cognitive radio capabilities start to proliferate, ones which can make very arbitrary decisions on modulation schemes and other transmission parameters, then it will be useful to consider more flexible detection schemes which do not rely on well-defined channelizations.

In Chapter 4, an extended energy detector is described. By expanding upon some

concepts associated with a traditional energy detector, the described algorithm is able to provide additional detection information about the spectral boundaries of a detected signal transmission within a bandwidth of interest. Since not only the occupancy decision itself, but also these boundaries, are also subject to detection error*, some of the terminology that defines the performance of an energy detector is extended and refined accordingly. The performance of this detector is also compared to a hypothetical detector which has perfect a-priori knowledge about the PSD characteristics of the signal being detected, with only the spectral boundaries being unknown. This is not a construct that is possible to realize in practice, but it serves as an* upper bound *on detection performance. Hence, it also provides a point of reference for comparing the performance of the described detector, as well as for further work relating to alternative arbitrarily channelized signal detection schemes.*

1.3.4 Exploiting Channel Occupancy Measurements to Guide the Selection of Channels to Utilize

Even in 'crowded' radio environments with apparently few spectral opportunities available for exploitation by a cognitive radio terminal, it is highly likely that there will be many channels to select from when deciding which one to utilize. It is an easy decision to make when *white* and *black* spaces are identified ('utilize' and 'avoid', respectively). However, the decision is less obvious for the continuum of *grey* spaces in between, which are only occupied by a licensed network user for a fraction of the time. At a given time, occupancy is represented by a binary value: conventionally, 0 means 'unoccupied' and 1 means 'occupied'.

It has been mentioned in section 1.2.4 that solving this problem requires two ingredients: a way of *characterizing* grey-space channels allowing them to be usefully compared, and a way of determining a *policy for selecting* grey-space channels based on these characterizations. Section 1.2.4 also described how [42] attempts to address both of these points, as well as some of the limitations encountered. Particularly, the authors determine what is the optimal channel selection policy in two cases: an environment consisting of a moderate but highly predictable (low complexity) occupancy,

and an environment with moderate but highly unpredictable (high complexity) occupancy. Such radio environments were *homogeneous*, meaning that the channels that were available for the cognitive radio to exploit all had similar values for overall occupancy pattern duty cycle and complexity. Real-world radio environments will generally present a far more heterogeneous environment, with channels exhibiting a mixture of occupancy patterns which complicate the learning problem and the formulation of channel selection policies. [42] also suggests a correlation between the complexity of the occupancy pattern and the 'probability of successful Q-learning'. However, the authors do not clearly describe how they measured the probability of successful learning, or the exact criteria for declaring learning to be 'successful'. They also do not describe how a cognitive radio should treat channels for which Q-learning was not successful, once in operation.

In [59], channel occupancies are also represented over time by a sequence of binary values – the authors call this a *binary time series*. The focus here is to use historical occupancy values on a channel to construct a model for predicting future occupancy values based on a least-squares-regression approach. The conceptual utility of being able to predict future occupancy values on a channel is obvious. However, their results suggest that a static model would only be useful for very-near-field predictions into the future.

Other work has used reinforcement learning to achieve goals relating to signal detection and channel utilization. For example, [60] uses *actor-critic* methods rather than Q-learning. The problem formulation used here assumes a relatively small number of frequency bands to select from, but each band is decomposed into a relatively large number of sub-bands which could, for example, correspond with individual OFDM sub-carriers. Rather than a pure channel selection policy, the cognitive radio attempts to learn a policy for selecting one of the following actions:

- Transmit data in the current frequency band, and then perform signal detection to see if any licensed network users are currently utlizing any of the sub-bands of the current frequency band.

- Perform signal detection on another frequency band (*out-of-band detection*) to determine which of its sub-bands are vacant and occupied. This information is used for strategic decision-making.

- Perform a switchover to another frequency band.

The objective is to find and utilize a frequency band that presents the most number of vacant sub-bands for the cognitive radio over time. However, this objective presupposes that all vacant sub-bands are of equal 'quality'. The work presented in [19] also attempts to answer the problem of choosing between *exploitive* actions (continuing to transmit on the current channel) and *exploratory* actions (looking to see what other channels appear to be vacant, and possibly switching to them). However, these results are again obtained from a homogeneous radio environment where all licensed network users transmit with equal duty cycle, suggesting that the learned policies are rather coupled to the radio environment that was presented during the learning process.

In Chapter 5, techniques for employing reinforcement learning to obtain a channel selection policy will be explored. By characterizing channels in a general manner based on both duty cycle and Lempel-Ziv complexity, and basing the possible actions on all of the available characterizations, a learned channel selection policy becomes more generally applicable to any radio environment, regardless of the specific occupancy characteristics of the channels within it. This makes the approach useful for mobile devices or radio environments with time-varying characteristics. It is also noted that calculating Lempel-Ziv complexity is quite computationally demanding, and an interesting approach for re-calculating efficiently when there is only an incremental change to the occupancy sequence is presented. In addition, unlike the aforementioned published work described in this section, the benefit of a learned channel selection policy was shown:

- *Even when a cognitive radio performs imperfect signal detection; and*

- *With simulated radio environments constructed according to temporal and spectral models derived from extensive real-world occupancy measurements (e.g. [19] assumes the periods where licensed network devices are transmitting and idle have durations which follow a simple exponential distribution).*

2

Exploit: Distinguishing Signal Features for Efficient Detection

The fundamental task of signal detection underpins the spectrum-sensing function of cognitive radios performing dynamic spectrum access.

A signal detector, in this context, takes a signal observation from its radio environment, constrained to some defined bandwidth and, based upon the available evidence within the observation, seeks to answer the following question within acceptable error bounds: *Was a wireless device of interest transmitting a signal within the defined bandwidth when the observation was taken?* The *"within acceptable error bounds"* qualifier is important, as any practical signal detector will be subject to some probabilities of decision error.

Various approaches for signal detection have been proposed for cognitive radios,

with differing degrees of computational complexity and nature of the underlying assumptions. The need for wireless communications to support ubiquity in communication service and access to information on demand implies mobile devices which are untethered and powered by batteries. The computational complexity and power consumption of a mobile device are strongly correlated in software-based systems – hence, *there is much motivation to reduce the computational complexity of the signal detection task*, particularly if it must be carried out frequently and periodically to avoid interfering with licensed network users. It is well known that preserving battery level in wireless devices, that continue to incorporate a plethora of features and functionality, is a constant challenge.

However, for imperfect signal detectors, reducing computational complexity may result in an unfavourable trade-off being made elsewhere – it is not possible to optimally meet all desirable outcomes simultaneously. For example, basing a decision upon a long observation (with respect to time) will generally lead to a lower probability of an incorrect decision, but lead to a higher decision latency – yet it is highly desirable to minimize both in a cognitive radio implementation. Incorrect decisions will result in either failing to exploit a spectral/temporal opportunity to transmit, or interfering with another device. On the other hand, a high latency or *delay* in the decision may increase the probability of the cognitive radio taking an action that is actually no longer suitable because the factors that contributed to the decision no longer apply – an issue with devices operating in mobile, rapidly-changing environments.

In this chapter:

- The problem of signal detection is considered in more detail
- The engineering trade-offs that are encountered in practical signal detectors are described
- An overview of several signal detection approaches is presented
- Detection approaches that exploit *cyclostationarity* – the presence of certain periodicities found in information-bearing RF transmissions – in the time and frequency domains are described

- Techniques for implementing a cyclostationarity detector with reduced computational complexity, whilst still providing superior performance to energy detection, are examined

2.1 Principles of Signal Detection

2.1.1 Binary Detection Principles

A signal detector makes a decision with respect to the *occupancy* of a frequency band and a time window (observation time) of interest. In reality, one of the following decision hypotheses is true:

$$\begin{aligned} &H_0 : \text{No signal was transmitted within the observation time and frequency band;} \\ &H_1 : \text{A signal transmission occurred within the observation time and frequency band.} \end{aligned} \tag{2.1}$$

The detector makes a decision, based upon evidence within the observation, as to which hypothesis is the correct one – as there are only two, this is a *binary* detection problem. Let R denote the random variable representing the 'reality' of which hypothesis is correct:

$$\begin{aligned} &R \in \{0, 1\}, \\ &R = \begin{cases} 0 & \text{when } H_0 \text{ is true} \\ 1 & \text{when } H_1 \text{ is true.} \end{cases} \end{aligned} \tag{2.2}$$

The evidence within the observation is converted into a value known as a *test statistic*, denoted as λ, by passing the observation through some test-statistic generation function. Suppose that the observation, $s[n]$, is a discrete-time representation of the *complex envelope* of a radio-frequency (RF) signal, with a length of N samples. It

could be considered to be a vector of length N:

$$\mathbf{s} = s[n] \text{ for } n = 0, \ldots, N-1. \tag{2.3}$$

The test statistic generation function can therefore be denoted:

$$\lambda = T(\mathbf{s}). \tag{2.4}$$

Different signal detection algorithms or approaches, fundamentally, are differentiated by their generation function $T(\mathbf{s})$. $s[n]$ is complex and has both *in-phase* and *quadrature* components:

$$\begin{aligned} s_I[n] &= \text{Re}\{s[n]\} \\ s_Q[n] &= \text{Im}\{s[n]\} \\ s[n] &= s_I[n] + j s_Q[n]. \end{aligned} \tag{2.5}$$

(2.1) can be rewritten more formally as:

$$\begin{aligned} H_0 &: s[n] = s_n[n] \\ H_1 &: s[n] = s_t[n] + s_n[n], \end{aligned} \tag{2.6}$$

where $s_n[n]$ is an *additive white Gaussian noise* (AWGN) component, and $s_t[n]$ is a component representing the signal of interest received by a cognitive radio (and transmitted by a licensed network user).

In practice, discrete-time signals are stochastic in nature, with each sample's value being subjected to random variability, due to the nature of data (usually assumed to be random) being transmitted by a device of interest ($R = 1$), and perturbations such as ambient, thermal and electrical noise (both cases when $R = 0$ or $R = 1$). As a result, λ is, for a given observation, a random variable whose *probability density function* (pdf)

is conditioned on R.

The decision is made by comparing λ to a *detection threshold*, λ_T. Let this decision be D; its value is determined as follows:

$$\begin{aligned} &D \in \{0, 1\} \\ &D = \begin{cases} 0 & \text{if } \lambda < \lambda_T, \ \therefore \text{Detector decides that } H_0 \text{ is true.} \\ 1 & \text{if } \lambda \geq \lambda_T, \ \therefore \text{Detector decides that } H_1 \text{ is true.} \end{cases} \end{aligned} \tag{2.7}$$

A *detection error* occurs when:

$$D \neq R. \tag{2.8}$$

There are two kinds of detection error (as mentioned previously in section 1.1.2) – they are repeated here but more formally:

- **False alarm**: Occurs when $D = 1$ and $R = 0$. The *probability of false alarm* is defined as:
$$P_{fa} = \int_{\lambda_T}^{\infty} f(\lambda|R = 0)d\lambda. \tag{2.9}$$
- **Missed detection**: Occurs when $D = 0$ and $R = 1$. The *probability of missed detection* is defined as:
$$P_{md} = \int_{-\infty}^{\lambda_T} f(\lambda|R = 1)d\lambda. \tag{2.10}$$

$f(\lambda|R)$ is the pdf of the test statistic λ, conditioned on R having a certain value.

A *perfect detector* is one where $P_{fa} = P_{md} = 0$. In other words, $\exists\lambda_T$ such that this is the case, when calculated according to (2.9) and (2.10). Graphically, this is saying that *pdfs* $f(\lambda|R = 0)$ *and* $f(\lambda|R = 1)$*, as functions of* λ*, do not overlap*, and λ_T *lies somewhere on the non-overlapping interval in between the two pdfs.* This is illustrated in Figure 2.1.

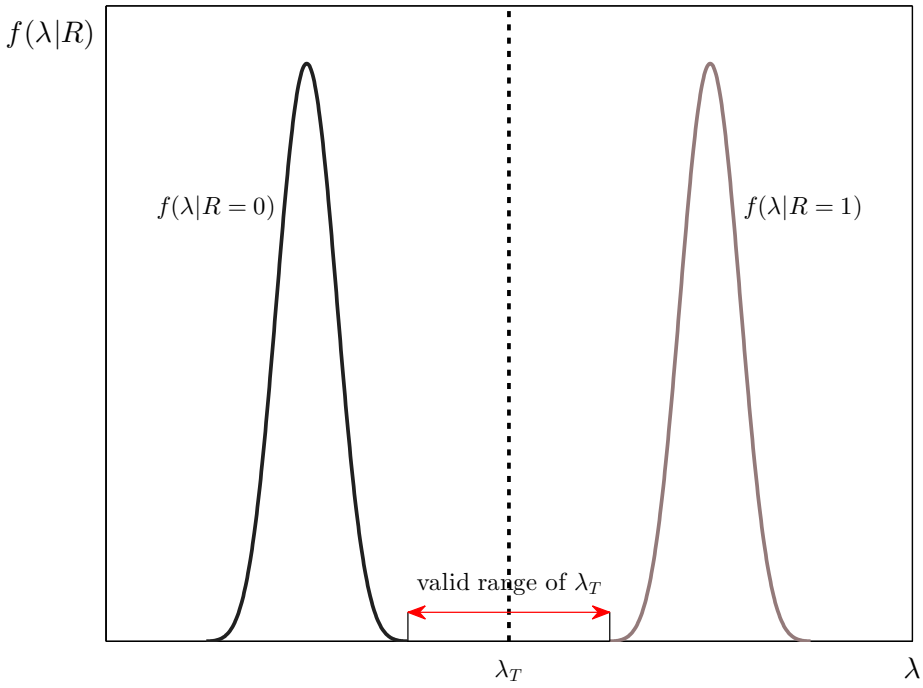

FIGURE 2.1: A perfect detector is possible where the pdfs $f(\lambda|R=0)$ and $f(\lambda|R=1)$ do not overlap.

In practical signal detectors, however, the pdfs will overlap, and there is no λ_T that will result in $P_{fa} = P_{md} = 0$ – see FIGURE 2.2.

It can be seen that, as λ_T is varied, a trade-off generally takes place between P_{fa} and P_{md} – decreasing one generally increases the other, even though it is desirable to have both as close to zero as possible. A *detector performance curve* shows how P_{fa} and P_{md} change as λ_T varies for a given detector implementation. Examples are illustrated in FIGURE 2.3 and FIGURE 2.4. *Each individual curve represents a particular detector implementation, and each point on a curve corresponds to a particular value of* λ_T *when applied to its corresponding detector implementation.*

In a practical detector, purely from the point of view of minimizing P_{fa} and P_{md} irrespective of other considerations, it is desirable to aim for a performance curve as close to that of an ideal detector as possible – the closer it is, the better the detector. From the detector performance curves, it can be seen that when one detector has better performance than another:

- It will have a lower value of P_{md} for a given value of P_{fa} (and vice versa).

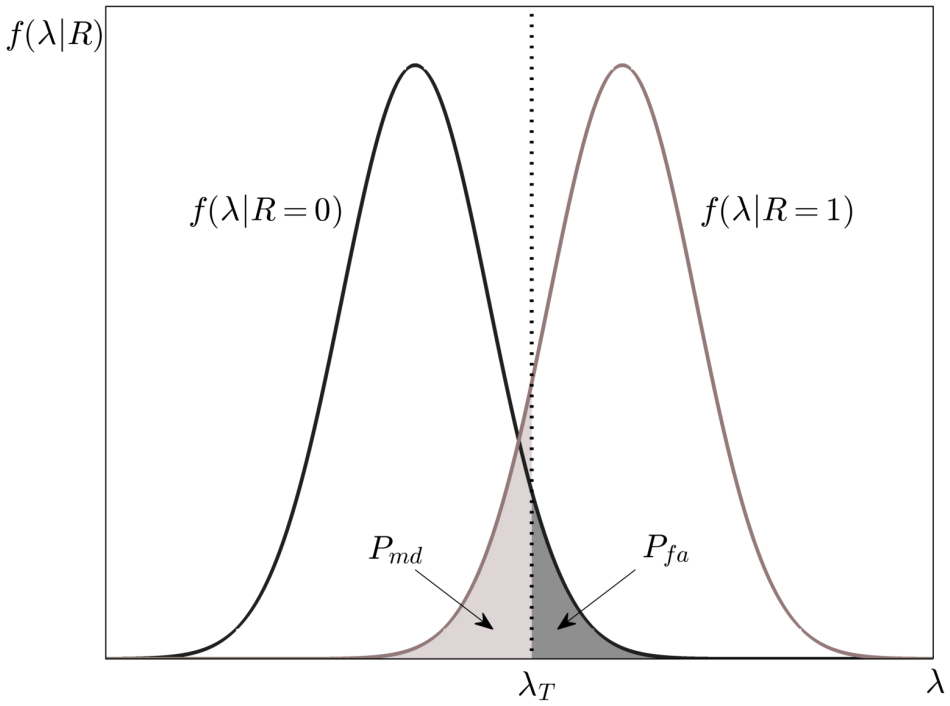

Figure 2.2: An imperfect detector, with non-zero probability of false alarm (P_{fa}) and probability of missed detection (P_{md}).

- When P_{fa} is reduced from one value, P_{fa_1}, to another, P_{fa_2}, the corresponding increase in P_{md} (given by $P_{md_2} - P_{md_1}$) will be smaller. A similar observation applies when reducing P_{md}, resulting in a smaller increase in P_{fa}. In other words, *the detection performance trade-off between P_{fa} and P_{md} is more favourable.*

Referring back to Figure 2.2, in terms of the conditional pdfs $f(\lambda|R = 0)$ and $f(\lambda|R = 1)$, these improved performance outcomes are achieved by:

- 'Spreading' the two pdfs further apart (i.e. there is a larger difference between their mean values) – assuming that the distribution of λ about the means for both pdfs (and hence the variance associated with both of them) remains unchanged; and/or

- 'Compressing' the distribution of the two pdfs towards their respective means (i.e. their variance is reduced) – assuming that their corresponding mean values remain unchanged.

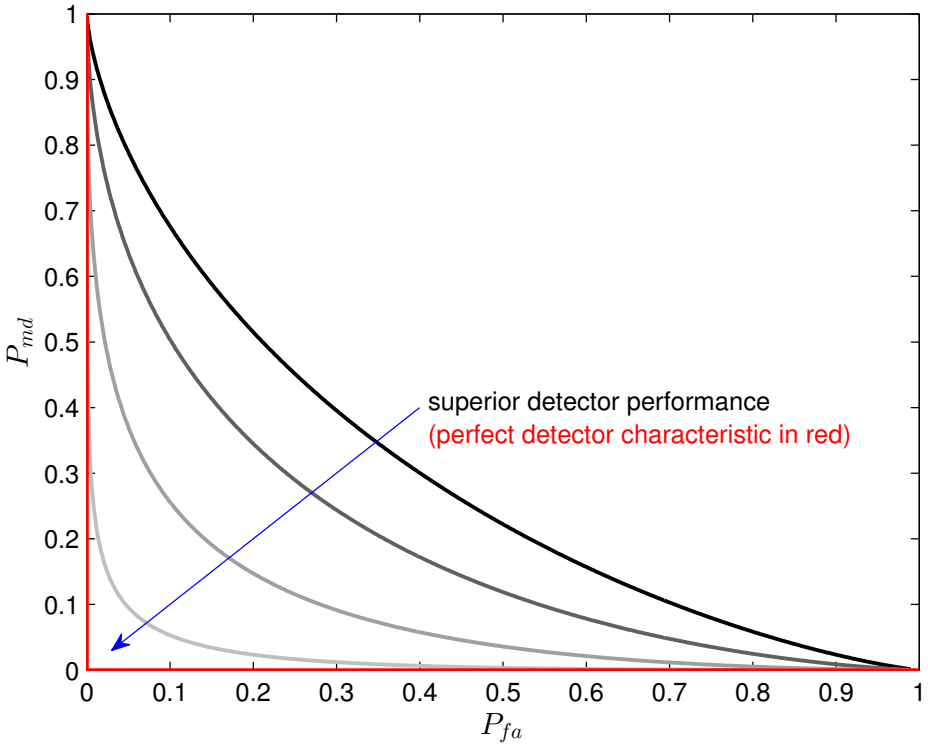

FIGURE 2.3: A series of detector performance curves shown with linear scales on both the P_{fa} and P_{md} axes.

2.1.2 The Resources of a Signal Detector

In order to improve detection performance curves, various 'resources' can be allocated to the signal detection problem. Applying these resources will affect the pdfs favourably as described in section 2.1.1 – however, they will generally adversely affect some other desirable outcome associated with the signal detector. These trade-offs are described in TABLE 2.1, and in the following subsections.

Knowledge

When detailed knowledge about the traits or *features* of a signal transmission to be detected is known in advance, these can then be used as strong discriminators to be incorporated into the detection algorithm, thus allowing for superior detection performance compared to approaches that do not exploit such knowledge.

An example of this is *waveform detection*, which exploits known patterns that are in the signal of interest, such as regularly-appearing pilot sequence patterns, compared

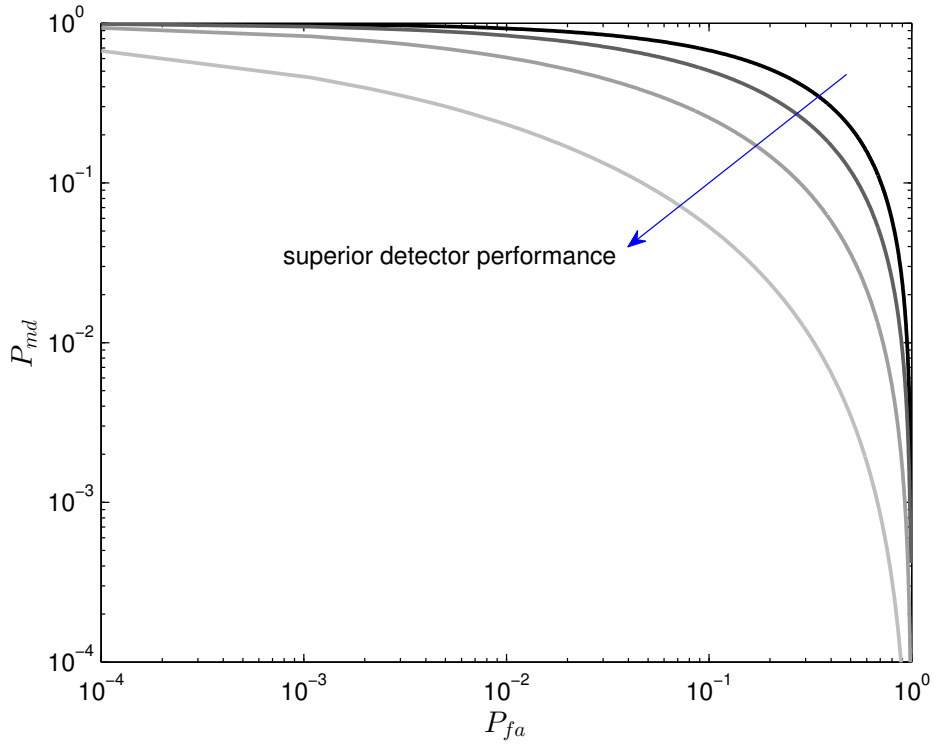

Figure 2.4: A series of detector performance curves shown with logarithmic scales on both the P_{fa} and P_{md} axes.

to *energy detection*, which simply uses the energy in the signal and no other characteristic. In [24], it is shown that waveform detection results in improved detection error performance, for a given observation time, compared to energy detection, and that the latter is particularly problematic in low-SNR environments. However, waveform detection is only useful for radio environments where the transmissions of the licensed users have these known patterns, such as DTV [23].

Waveform detectors are basically, by their nature, 'single-purpose' and inflexible – they are good for detecting signals that exhibit a known pattern, but not for detecting anything else. A cognitive radio would have to implement a bank of waveform detectors for every possible pattern that it might encounter in the field and consider the value of the test statistic λ resulting from each.

Alternatively, it would have to incorporate *machine-learning* capabilities, and incur the cost of applying this in an attempt to learn about what known recurring patterns might exist in the RF signals seen in a given radio environment, in order to improve

Resource	Description	Cost of Improved Detection Performance
Knowledge	Knowledge or information about the signal to be detected that distinguishes it from other signals	Loss of flexibility/generality; machine learning needs to be performed
Computation	Perform transformations on a signal to discover more about it and extract 'better' features useful for detection (analysis)	Increased power consumption; increased detection latency
Time	Spend longer observing a signal before making a detection decision	Increased detection latency; increased energy consumption

TABLE 2.1: The costs of allocating various resources to the signal detection problem.

detection error performance.

Computation

In principle, additional *analysis*, and its associated computation, can be performed on a signal observation in order to uncover features or characteristics that would not otherwise be apparent, which can in turn be used to improve detection performance. A contrived example of this might be the case of a sinusoidal signal buried in noise – it may not be apparent when looking at the time-domain representation that there is a distinct component of a certain frequency there, but performing frequency-domain analysis such as using the DFT (Discrete Fourier Transform) would uncover this. A related and more pertinent example for cognitive radio is *cyclostationarity* detection, which uses additional analysis to uncover the presence of otherwise-hidden periodicities in a signal observation, compared to, say, energy detection [23].

In computer architectures, and particularly SDR, the amount of computation in an algorithm can be measured by the number of processor *instruction cycles* (clock cycles where an instruction is executed by a processor) that would be required to complete it in software. However, this is highly dependent on an SDR's actual underlying processor architecture and instruction set, memory and cache architecture, the precision with which signal values are stored in memory (word width, fixed point or floating

point), as well as how well optimized the software implementation of the algorithm is when compiled or assembled. More abstractly, and for comparison between different algorithms, the amount of computation may also often be expressed in terms of the number of primitive arithmetic operations involved, such as *real multiplies* (multiplying two real numbers) or *complex multiplies* (multiplying two complex numbers – a complex multiply generally involves four real multiplies).[1]

Performing additional computation comes at the cost of overall detection *latency* (the time elapsed between when the collection of a signal observation is commenced, and the corresponding detection decision being made), and/or more power consumption:

- If latency is preserved, then additional computation has to be performed in the same amount of time, implying more processor instruction cycles being required per unit time, and hence a higher power consumption for the duration of the computation
- If power consumption is preserved (by issuing the same rate of instructions to the processor), the additional computation will take longer to complete, leading to higher detection latency

Time

In general, a detector calculating a test statistic λ on a longer observation window will result in improved detection-error performance - in [24], this is demonstrably true for both energy and waveform detection.

This comes at a cost of detection latency (the act of collecting a signal over a longer window takes longer and delays the execution of the detection algorithm), and increased energy consumption (the detection algorithm has to process more signal and hence will require more processor cycles to do so). In [24], detailed results in relation

[1] In the past, many processor architectures required more clock cycles to perform multiplication than other arithmetic operations such as addition, leading to many computational cost models being largely based on a count of the number of multiplication operations required. This discrepancy is less true for many modern processors, although it is the author's experience that often, the *latency* between the issuing of a multiplication instruction and its result being available for other instructions is still larger than for other operations.

to this trade-off between detection-error performance and latency associated with long observation windows, for both energy and waveform detection, are presented – the longer the window, the lower the probability of sensing error.

2.2 Types of Signal Detectors

Here, a summary of various signal detection approaches that have been proffered for cognitive radio applications is presented.

Energy detection is the simplest detection algorithm and the least computationally complex, but its error performance is inferior to other approaches.

Waveform detection requires a known pattern in the signal to be known in advance, but exploiting this knowledge allows it to achieve superior error performance where this criterion is met.

Cyclostationarity detection[2] is also capable of offering better error performance compared to energy detection (but inferior to waveform detection), but is much more computationally demanding.

Energy detection, which uses the least amount of information about a signal, represents a *lower bound* on the possible detection performance of a signal detector (i.e. highest probability of detection error). Conversely, waveform detection uses a complete description of a signal, and therefore the most information about it, and represents an *upper bound* on detection performance (i.e. lowest probability of detection error) [24]. Other detection schemes lie somewhere within these bounds in terms of detection-error performance.

2.2.1 Energy Detection

In concept, an energy detector simply uses the total energy in a signal observation as a test statistic. The sensed signal observation is passed through a squaring function,

[2]Existing literature often refers to *cyclostationary detection*. The author prefers to use "cyclostationarity" instead, as it is the signal property that the detection process measures, whereas "cyclostationary" is an adjective describing the signal observation, not the detection process itself.

the result of which is then passed through an integrator. For discrete-time signals, this can be represented in its basic form as:

$$\lambda = \sum_{n=0}^{N-1} |s[n]|^2. \tag{2.11}$$

On average, a signal observation will have more energy when $R = 1$, compared to when $R = 0$, which is the principle of operation – the threshold λ is placed somewhere between these two average energy values, depending on the constraints imposed on P_{fa} and/or P_{md}.

An alternative expression which can be used for the test statistic is:

$$\lambda_{norm} = \frac{2}{\sigma_n^2} \sum_{n=0}^{N-1} |s[n]|^2, \tag{2.12}$$

where σ_n^2 is the complex noise power, which is generally considered to be divided equally between the real (in-phase) and imaginary (quadrature) parts of a complex signal. This formulation is based on the results presented in [61].

The difference between (2.11) and (2.12) is simply a scaling factor based on the prevailing noise power. However, for a signal detector to employ (2.12), this noise power must be known in advance (or estimated accurately). The advantage of (2.12) is that λ_{norm} follows a χ^2-distribution (based upon *standard normal distributions*) [61] – more particularly:

- When $R = 0$, λ_{norm} follows the central χ^2-distribution with $2N$ degrees of freedom (DOF)

- When $R = 1$, λ_{norm} follows the non-central χ^2-distribution with $2N$ DOF, and non-centrality parameter $\delta = 2N \times \text{SNR}$

SNR is defined as the ratio between the average signal power of $s_t[n]$ and the average signal power of $s_n[n]$ in an observation, as that seen by the cognitive radio performing

spectrum sensing:

$$\rho \triangleq \text{SNR} = \frac{\frac{1}{N}\sum_{n=0}^{N-1}|s_t[n]|^2}{\sigma_n^2}. \tag{2.13}$$

Sections A.1 and A.2 provide finer details about these assertions.

Applying Parseval's Theorem to (2.11) yields an equivalent formulation from the frequency-domain representation of $s[n]$:

$$\lambda = \frac{1}{N}\sum_{k=0}^{N-1}|S[k]|^2, \tag{2.14}$$

where $S[k]$ is the kth DFT coefficient of $s[n]$.

2.2.2 Waveform Detection

Waveform detection requires prior knowledge about the waveform (or a subset of it) that will be present in a transmission (albeit corrupted by noise) from a licensed network user – in other words, $s_t[n]$ is known. A signal observation is then correlated with the known waveform to obtain the test statistic [62]:

$$\begin{aligned}\lambda &= \text{Re}\left\{\sum_{n=0}^{N-1} s[n]s_t^*[n]\right\} \\ &= \begin{cases} \text{Re}\left\{\sum_{n=0}^{N-1} s_n[n]s_t^*[n]\right\}, & \text{when } R=0 \\ \text{Re}\left\{\sum_{n=0}^{N-1} s_n[n]s_t^*[n]\right\} + \sum_{n=0}^{N-1}|s_t^2[n]|^2, & \text{when } R=1. \end{cases}\end{aligned} \tag{2.15}$$

Conceptually, it is the extra term representing the energy in $s_t[n]$, $\sum_{n=0}^{N-1}\left|s_t^2[n]\right|^2$, that provides the 'separation' between the means of the pdfs $f(\lambda|R=0)$ and $f(\lambda|R=1)$.

2.2.3 Cyclostationarity Detection

Cyclostationarity detection identifies the presence of certain periodicities in a signal observation that would be present in an information-bearing communication signal (say, from a licensed network user) [63], as well as signals in other artificial environments such as radar [64] and telemetry systems, but which would be absent from one in which only AWGN was present.

A cyclostationarity detector must therefore, conceptually:

- Determine the presence and nature of the cyclostationarity *features* of $s_t[n]$ when it is present in a channel
- Calculate a test statistic based on the features, such that it has a high value when the features are present, and a low value when they are absent (due to H_0 being true).

Some concepts relating to cyclostationarity are now described.

Definitions of Cyclostationarity

A signal is cyclostationary if it has a statistical parameter which varies periodically with respect to time [63]. Because artificial, information-bearing signals are randomly-varying (stochastic) in nature rather than deterministic, their statistics cannot be considered absolutely – rather, they must be considered *in the average* when determining the presence of cyclostationarity.

A continuous-time signal that is periodic has one or more non-zero Fourier Series coefficients. Conceptually, for a deterministic, periodic signal $s(t)$, there are distinct frequency components at one or more frequencies $\alpha = kf_0 = \dfrac{k}{T_0}$, where k is an integer, f_0 is the fundamental frequency and $T_0 = \dfrac{1}{f_0}$ is the fundamental period. The exponential Fourier Series coefficients associated with these frequencies is given by:

$$s_\alpha = \frac{1}{T_0}\int_0^{T_0} s(t)e^{-j2\pi\alpha t}dt. \tag{2.16}$$

Note that the expression of the Fourier Series coefficients is equivalent to a time-averaging operation over the signal's period, T_0. Strictly speaking, performing the integration over any time period that is an integer multiple of T_0 will yield the same result. However, by extending the time-averaging window towards infinity, this condition vanishes as the result converges towards the true value:

$$s_\alpha = \lim_{T\to\infty} \frac{1}{T} \int_0^T s(t) e^{-j2\pi\alpha t} dt. \tag{2.17}$$

For stochastic signals that are ergodic, (2.17) is used to determine the underlying Fourier Series coefficients.

For a signal to have *first-order cyclostationarity*, there needs to be one or more underlying non-zero, sinusoidal components, so that its ensemble average is periodic in time. This is equivalent to saying that there is at least one non-zero value of α such that its corresponding Fourier Series coefficient is also non-zero:

$$\exists\, \alpha \neq 0 : s_\alpha \neq 0. \tag{2.18}$$

Let the *delay product* of a signal, $s(t)$, represent the product of two time-translated versions of itself, with a total delay of τ between them [63]:

$$D(t,\tau) = s\left(t + \frac{\tau}{2}\right) s^*\left(t - \frac{\tau}{2}\right). \tag{2.19}$$

Using a similar definition as in (2.17), the Fourier Series coefficients of the delay product of a signal can be defined on a set of frequencies $\alpha = \dfrac{k}{T_0}$, in the case where it is periodic in the average (with fundamental period T_0):

$$s_\alpha^2(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_0^T s\left(t + \frac{\tau}{2}\right) s^*\left(t - \frac{\tau}{2}\right) e^{-j2\pi\alpha t} dt. \tag{2.20}$$

A signal has *second-order cyclostationarity* if its delay product is periodic in time, for some delay values of τ. In other words, for some values of τ, there is at least one

non-zero coefficient in the Fourier Series of the delay product:

$$\text{For some } \tau, \exists\, \alpha \neq 0 : s_\alpha^2(\tau) \neq 0. \tag{2.21}$$

The values of α indicated in (2.18) and (2.21) are referred to as the *cycle frequencies*. They are integer multiples of a *fundamental* cycle frequency, f_α, which is equal to the reciprocal of the delay product's period:

$$\alpha = m f_\alpha, \tag{2.22}$$

where m is an integer representing the Fourier Series coefficient index.

It can be seen that (2.20) can be considered as a function of two variables α and τ – at every α, the integral varies with τ. This is called the *cyclic autocorrelation* function (CAF) of $s(t)$ [63]:

$$R_s^\alpha(\tau) \triangleq s_\alpha^2(\tau). \tag{2.23}$$

Note that *when $\alpha = 0$, the definition of CAF given by (2.20) reverts to the usual definition of the autocorrelation function of a complex signal with finite power.*

By taking the Fourier Transform of the cyclic autocorrelation function (with respect to varying τ but fixed α), the result is called the *spectral correlation density*[3] (SCD) [63]:

$$S_s^\alpha(f) = \int_{-\infty}^{\infty} R_s^\alpha(\tau)\, e^{-j2\pi f \tau} d\tau. \tag{2.24}$$

The result is that the SCD is effectively a function of two variables α and f. This relationship – sometimes referred to as the cyclic Wiener relation – is analogous to the relationship defined by the Wiener–Khinchin theorem [67], which states that the power spectral density of a signal is equal to the Fourier Transform of its autocorrelation function. Furthermore, since the CAF is equal to the standard autocorrelation function

[3]The term *cyclic spectrum* appears to be used in some published works (e.g. [65], [66]). In [63], the author indicated that he has also used the term *cyclic spectral density* in earlier work. On the other hand, the term *cycle spectrum* is used in [63] to mean the set of cycle frequencies over all τ.

when $\alpha = 0$, it follows from the Wiener-Khinchin relation and (2.24) that *when $\alpha = 0$, the definition of the SCD reverts to that of the power spectral density of the signal.*

In [63], an intuitive explanation of the SCD is offered – for a given frequency f and cyclic frequency α, it is the average signal power that remains when $s(t)$, passed through an infinitesimally narrow bandpass filter (BPF) centred at $f - \frac{\alpha}{2}$, is multiplied by the conjugated result of $s(t)$ passed through an infinitesimally narrow bandpass filter centred at $f + \frac{\alpha}{2}$. Alternatively, two identical bandpass filters centred at f can be used, with spectral shifting of $s(t)$ carried out by multiplying by a complex exponential function. These are shown in FIGURE 2.5.

From this, an intuitive explanation of 'cyclic frequency' also emerges – a non-zero component in the SCD at cyclic frequency α means that there are two spectral components to the signal, spaced apart by α Hz and with frequency f Hz as the midpoint, that are correlated over time. OFDM signals with pilot subcarriers are an example of a class of signals that readily exhibit such correlations.

It is also worth noting that whilst continuous-time signals have been used to describe the above definitions, analogous definitions can apply for discrete-time signals, by employing the DFT in lieu of the Fourier Series, and the discrete-time Fourier Transform (DTFT) or DFT in lieu of the Fourier Transform. However, an observation is that delay values are constrained to an integer multiple of the sampling period, T_s, so the halving implied in (2.20) does not work as it requires values in between those which are available. However, because the limits of integration approach infinity, it is possible to time-shift the entire integrand by $\frac{\tau}{2}$ without affecting the result:

$$\begin{aligned} R_s^{\alpha}(\tau) &= \lim_{T\to\infty} \frac{1}{T} \int_0^T s(t+\tau)\, s^*(t)\, e^{-j2\pi\alpha\left(t+\frac{\tau}{2}\right)} dt \\ &= e^{-j\pi\alpha\tau} \left(\lim_{T\to\infty} \frac{1}{T} \int_0^T s(t+\tau)\, s^*(t)\, e^{-j2\pi\alpha t} dt \right). \end{aligned} \tag{2.25}$$

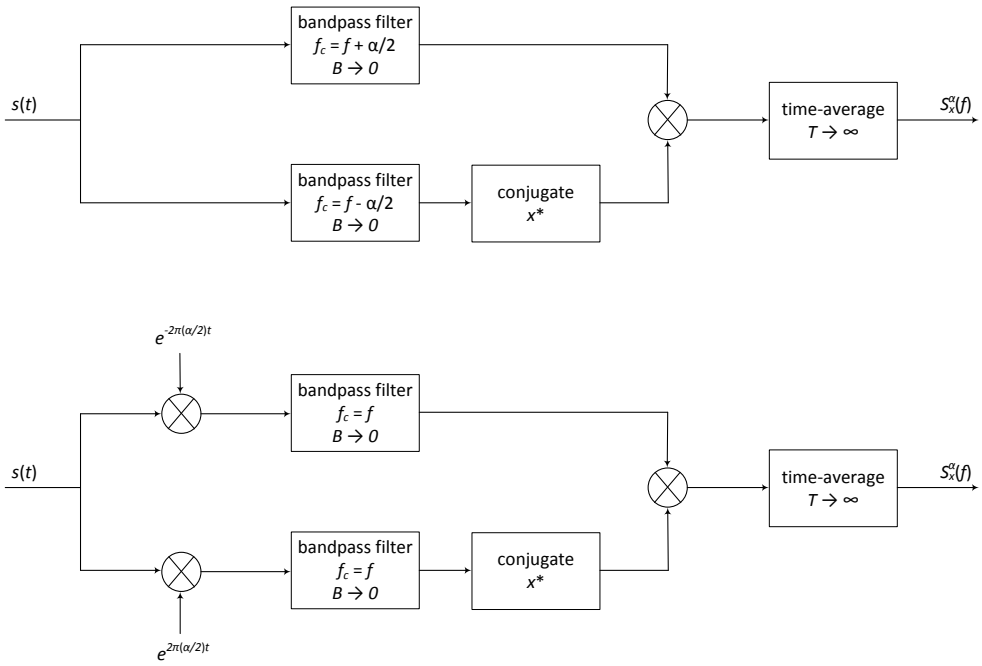

FIGURE 2.5: Equivalent ways of measuring the SCD. f_c represents the centre frequency and B the bandwidth of the bandpass filter, while T is the averaging interval.

The discrete-time counterpart to this then becomes [63]:

$$R_s^\alpha[\nu] \triangleq e^{-j\pi\alpha\nu T_s} \left(\lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} s[n+\nu] s^*[n] e^{-j2\pi\alpha n T_s} \right), \tag{2.26}$$

where ν is the number of delay samples, so $\tau = \nu T_s$.

2.3 Implementing a Cyclostationarity Detector

2.3.1 Estimating the Cyclic Autocorrelation

OFDM is a multi-carrier transmission scheme that is based on the inverse Discrete Fourier Transform (IDFT) to achieve orthogonality between the individual subchannels. The underlying binary data stream being transmitted is *demultiplexed* onto K logical subchannels, achieved using a K-point IDFT, so that each subchannel has a

carrier signal that is modulated separately, and whose frequency corresponds with an IDFT frequency bin. Rather than preventing intercarrier interference (ICI) through the use of non-overlapping subchannel bands with *guard bands* in-between, OFDM uses overlapping, orthogonal bands which, along with the elimination of the need for guard bands, results in a reduced bandwidth requirement overall. In effect, it converts a data stream into K parallel streams, each with a factor of K reduction in signaling rate, compared to what would otherwise be necessary (to achieve the same overall data rate). The slower signaling rate on each subchannel makes OFDM particularly robust against *multipath delay*, since any delay is much reduced when considered as a fraction of a subchannel's signaling period. Another notable feature of OFDM is the use of a *guard interval* between each OFDM symbol. This also eliminates *intersymbol interference* (ISI), as long as the maximum multipath delay is always smaller than the guard interval. Furthermore, where the guard interval is populated with a *cyclic prefix*, which contains a repetition taken from the end of the next OFDM symbol, ICI due to multipath delay can be largely mitigated, as the OFDM receiver is more likely to obtain the complete signal information within a symbol on each subchannel even if it is delayed by multipath effects. More details about OFDM are presented in [68].

An OFDM signal with a guard interval consisting of a cyclic prefix, is an example of one that exhibits second-order cyclostationarity. An OFDM symbol has two components in time – the guard interval (consisting of the cyclic prefix), and the *data-bearing* interval. If the data-bearing interval consists of N_{data} samples, the cyclic prefix consists of N_{cp} samples, where N_{cp} is some fraction of N_{data}. (If there is no guard interval, $N_{cp} = 0$.) The number of samples in one OFDM symbol is then N_{symb}:

$$N_{symb} = N_{cp} + N_{data} \tag{2.27}$$

and one symbol is represented in discrete time by the vector whose elements are time samples:

$$\mathbf{s}_{symb} = [\overbrace{s_{data}[N_{data}-N_{cp}], s_{data}[N_{data}-N_{cp}+1], \ldots, s_{data}[N_{data}-2], s_{data}[N_{data}-1],}^{\text{cyclic prefix}} \underbrace{s_{data}[0], s_{data}[1], s_{data}[2], \ldots, s_{data}[N_{data}-2], s_{data}[N_{data}-1]}_{\text{data-bearing component}}], \tag{2.28}$$

where $s_{data}[n]$ is the discrete-time representation of the data-bearing component of the OFDM symbol. In general, an OFDM signal, $s[n]$ of infinite duration can hence be expressed as the following sequence:

$$\begin{aligned} s[n] = \{&s_k[N_{data}-N_{cp}], s_k[N_{data}-N_{cp}+1], \ldots, s_k[N_{data}-2], s_k[N_{data}-1], \\ &s_k[0], s_k[1], s_k[2], \ldots, s_k[N_{data}-2], s_k[N_{data}-1], \ldots\}_{k=1}^{\infty} \end{aligned} \tag{2.29}$$

where $s_k[n]$ is the discrete-time representation of the kth symbol.

Such an OFDM signal has significant components in its CAF for $\tau = 0, \pm N_{data}T_s$ [69] (equivalently, $\nu = 0, \pm N_{data}$):

$$\begin{aligned} T_{symb} &\triangleq N_{symb}T_s \\ T_{cp} &\triangleq N_{cp}T_s \\ T_{data} &\triangleq N_{data}T_s. \end{aligned} \tag{2.30}$$

For $\tau = 0$, the delay product becomes:

$$D(t,0) = s(t)\, s^*(t) = |s(t)|^2 . \tag{2.31}$$

If $s(t)$ is an OFDM signal bearing random data, then the Fourier Series coefficients

which determine the CAF will contain a dominant value at $\alpha = 0$ (equal to the mean-square magnitude of $s(t)$ according to (2.20)), and zero elsewhere. As an aside, it is obvious that the delay product is a real-valued function even for complex $s(t)$, and hence so is the CAF for $\tau = 0$.

For $\tau = \pm T_{data}$, since the time elapsed between the cyclic prefix and its repetition in the corresponding data-bearing component is T_{data} for each symbol, the cyclic prefix in one term of the delay product (2.19) becomes time-aligned with its conjugate taken from the data-bearing component in the other term. In other words:

$$\begin{aligned} &\text{When } \tau = \pm T_{data}, \\ &s\left(t + \frac{\tau}{2}\right) = s\left(t - \frac{\tau}{2}\right), \\ &\text{for } kT_{symb} + \frac{T_{data}}{2} \leq t \leq kT_{symb} + \frac{T_{data}}{2} + T_{cp}, \text{ with integer } k. \end{aligned} \tag{2.32}$$

So, for the delay product:

$$\begin{aligned} &D\left(t, \pm T_{symb}\right) = s\left(t \pm \frac{T_{symb}}{2}\right) s^*\left(t \pm \frac{T_{symb}}{2}\right) = \left|s\left(t \pm \frac{T_{symb}}{2}\right)\right|^2, \\ &\text{for } kT_{symb} + \frac{T_{data}}{2} \leq t \leq kT_{symb} + \frac{T_{data}}{2} + T_{cp}, \text{ with integer } k. \end{aligned} \tag{2.33}$$

For other values of t, assuming that random data transmission occurs, D has an expected value of zero. The delay product therefore has relative peaks that are spaced apart by T_{symb}, giving rise to non-zero Fourier Series coefficients which determine the CAF.

FIGURE 2.6 provides an illustrative example of what the *magnitude* of the CAF may look like for an OFDM signal. Note that, for ease of estimation with discrete-time signals, the Fourier Series coefficient evaluation operation in (2.20) was replaced by a DFT over a finite-length, discrete-time signal observation. Assuming that the observation has a length equal to M symbols, the kth DFT coefficient can be considered to be an approximation for the Fourier Series coefficient corresponding to α under the

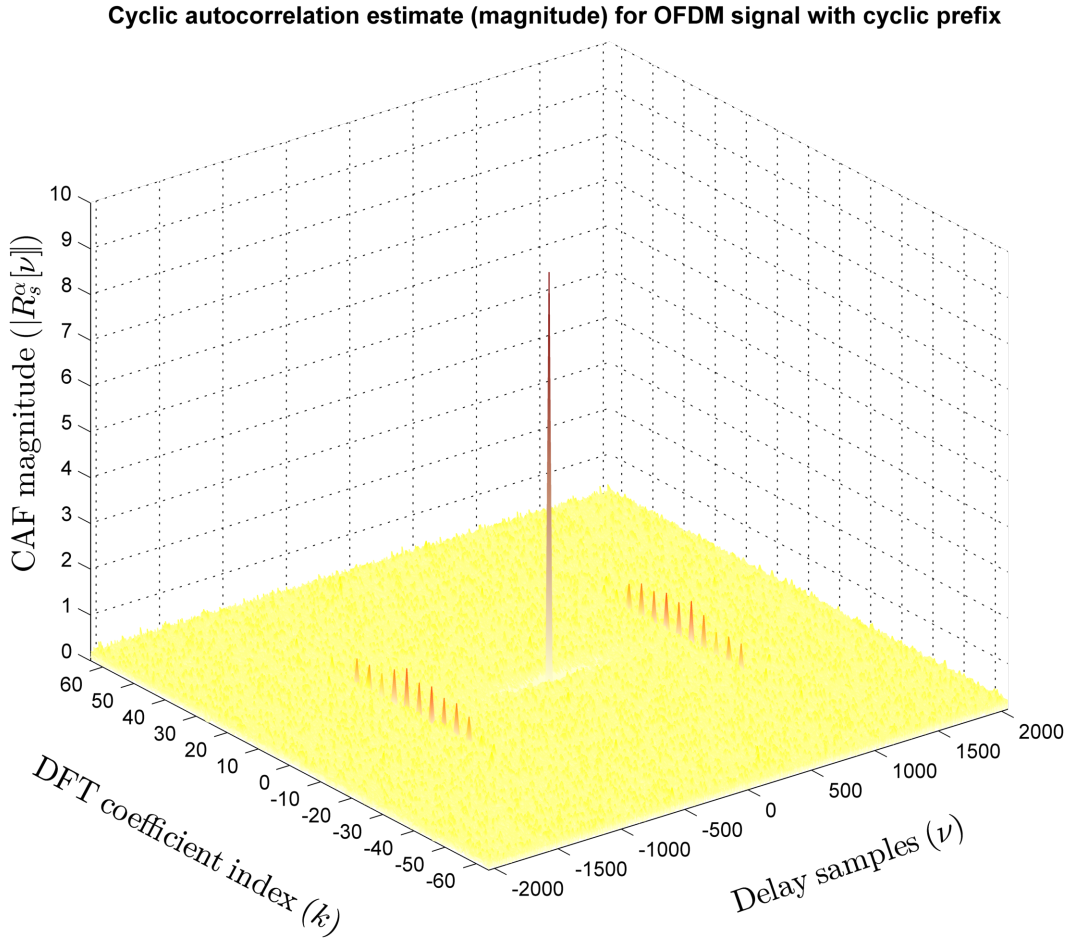

Figure 2.6: An estimate of the CAF (magnitude only) for an OFDM signal. The observation length used for the computation was four symbols long. $N_{cp} = \frac{1}{8} \times N_{data}, N_{data} = 1024$. The large peak at $\nu = 0, \alpha = 0$ predicted by (2.31) is apparent, as is the line of peaks at $\nu = \pm N_{data}$.

condition:

$$k = \alpha M T_{symb}. \tag{2.34}$$

(Refer to section A.3 for a justification of this.) Also, due to the finite length of the signal observation used, a *circular shift* operation by ν samples was carried out instead of a linear shift (which would have required twice as many samples).

A signal detector that exploits the CAF may require a very long signal observation of at least several symbols in order to capture the cyclic prefix, and thus may be subject to a larger detection latency. This is accentuated when it is not known in

advance how long the symbol durations are likely to be, so a conservative approach of adopting a longer observation length would need to be employed. Furthermore, it is only useful for detecting licensed network transmissions that employ modulations with regular time-based repetition such as OFDM with a cyclic prefix.

2.3.2 Estimating the Spectral Correlation Density

The definition of SCD in (2.24) relies on an infinite integral on a continuous-time signal, yet, like for the CAF, for cognitive radio applications it is highly desirable to be able to estimate the SCD based on a discrete-time signal observation of finite length.

One such approach is called the *FFT Accumulation Method* (FAM), which employs heavily the use of the Fast Fourier Transform (FFT) for computational efficiency, and hence lends itself neatly to the problem of estimation using finite-length, discrete-time observations.

Conceptually, and referring to FIGURE 2.5, some of the main ideas behind FAM are:

- Rather than implementing a bank of narrow bandpass filters with every possible centre frequency of interest, the average component at each centre frequency is estimated via a single FFT operation on a contiguous subset, of length N' samples, of the observation (which has N samples; $\therefore N' < N$). The effective bandpass-filter bandwidth is then equal to the frequency resolution of the FFT operation:
$$B = \Delta f = \frac{f_s}{N'}. \tag{2.35}$$

- The output of each bandpass filter as a time sequence of samples (referred to as a *complex demodulate* in [70]) can be modelled by the sequence of FFT coefficients at the corresponding frequency bins obtained by repeated FFT operations on windows of length N', where each window 'slides across' the observation, and whose starting point is circular-shifted from that of the previous window by one sample.

- To reduce computational complexity, FAM allows for *decimation* of the complex demodulates by making each window circular-offset from the previous window by L samples, where $1 < L < N'$ – effectively reducing the length of each complex demodulate sequence by a factor of L. ($L = 1$ implies no decimation.) The number of such windows is P, which is also the length of each decimated complex demodulate sequence:

$$N_{fam} = PL, \tag{2.36}$$

  where N_{fam} is the number of samples required to perform a single SCD estimate using FAM.

- For a given f, rather than applying the averaging block individually on a separate sequence corresponding to each α value, it is possible to obtain estimates at P adjacent α values through a single P-point FFT operation, as long as the α values of interest are discretized to the following resolution:

$$\Delta\alpha = \frac{f_s}{N_{fam}}. \tag{2.37}$$

  The P values used in this FFT correspond to the conjugate products of decimated complex demodulate sequences obtained from the first FFT stage. Furthermore, the P adjacent α values for which estimates are obtained this way are enclosed within defined *channel-pair regions* [70].

FAM is described in detail in [70] and [64].

It is noteworthy that FAM, as applied to discrete-time signals, has a domain encompassing a diamond-shaped region on the $f - \alpha$ plane, as illustrated in Figure 2.7, since for a given f, α is constrained by the Nyquist frequency such that $\left| f \pm \left| \frac{\alpha}{2} \right| \right| \leq \frac{f_s}{2}$.

Finally, FAM gives a *discretized* SCD estimate, with *frequency resolution* given by (2.35), and *cycle frequency resolution* given by (2.37).

In later work presented in this thesis, cyclostationarity detection based on SCD estimates obtained via FAM are illustrated - this approach is used in preference to

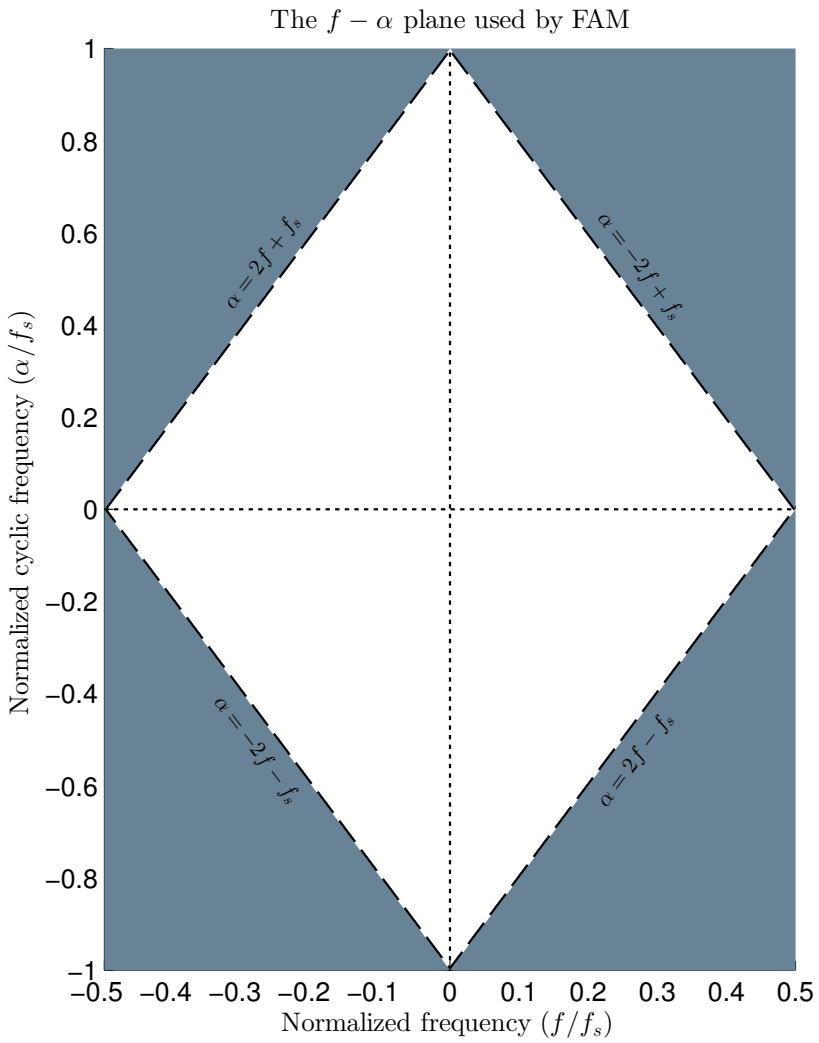

FIGURE 2.7: The domain of the SCD estimate given by the FFT accumulation method (FAM), in white.

exploiting the CAF due to the latter's applicability to limited modulation types.

2.4 Cyclostationarity Detection as a Superset of Energy Detection

As elaborated within section 2.2.3, the SCD gives the PSD of a signal for $\alpha = 0$. Since, by definition, integrating the PSD over frequency gives the signal power (energy normalized by time), it is apparent that *a signal detector that relies on the SCD will have at least as much information available about the signal as an energy detector.*

Therefore, cyclostationarity detection may be considered to be a superset of energy detection – *therefore, any cyclostationarity detector exploiting at least the complete CAF or SCD definition at* $\alpha = 0$ *will, by definition, be able to achieve detection-error performance which is at least as good as that of energy detection.* Any improvement resulting in superior detection performance is due to exploiting signal information corresponding to $\alpha \neq 0$.

2.5 Cyclostationarity Detector: Exploiting Knowledge to Reduce Computation

The imperative for cognitive radios to operate efficiently in mobile devices, subject to limited battery power, has already been elucidated. The motivation here is to design a spectrum sensing technique based on cyclostationarity detection with a highly reduced computational complexity, whilst still providing for superior detection-error performance. The computational complexity here directly impacts upon the energy efficiency of the sensing technique implemented on battery-powered wireless devices. Having energy-efficient detection is crucial, especially when the battery level is low.

Cyclostationarity detection demonstrably involves high computational complexity if the entire CAF or SCD needs to be estimated. In [56], a cyclostationarity detector utilized a detection test statistic that performed an integration operation on an SCD estimate over the entire $f - \alpha$ plane. It was thus a very computationally-demanding process, both in terms of estimating the SCD and then calculating the necessary test statistic – and hence was not ideal for power-constrained devices.

Here, the ideas behind a sensing technique based on a cyclostationarity detection method are explored – it outperforms the energy detector in terms of detection-error performance, whilst simultaneously lowering computational complexity by only computing meaningful subsets of the SCD. The detection-error performance of the detector

is crucial for the minimization of interference to licensed network users in cognitive radio systems.

2.5.1 Improving Computational Efficiency

One measure to help reduce computational complexity involves employing FAM as an SCD estimator – this approach uses FFTs operating on overlapping windows taken from the signal observation.

Another measure involves exploiting *a priori* knowledge about the signal transmission that needs to be detected. For example, if it is known that a licensed network user is employing *square-root raised cosine* (SRRC) pulse shaping (or more accurately, a causal approximation thereof) in its transmission, $s_t[n]$, with some known symbol rate f_{symb}, then it is expected that, with a sufficiently large observation time, there will be a significant spectral 'signature' in the SCD, corresponding to cyclic frequency $\alpha = f_{symb}$. (SRRC pulse shaping is described in [71], and the ideal impulse response of a SRRC filter is derived in [72].) Note that the line where $\alpha = 0$, corresponding to the PSD of the signal observation, shows a characteristic 'bump' in the middle where the majority of the signal power is spectrally located. On the other hand, the SCD of AWGN (an absence of the signal) is flat across the entire $f - \alpha$ plane except for at $\alpha = 0$ – in addition, its PSD is also flat in comparison. These points are illustrated in FIGURE 2.8.

By exploiting this knowledge, a detector can be formulated which selectively estimates the SCD within certain regions of interest, based on spectral signatures expected from prior knowledge of the signal to be detected. FAM decomposes the spectral plane into a tiled arrangement of channel-pair regions [70] – SCD estimates within each channel-pair region can carried out independently of each other. Therefore, it is possible to selectively perform SCD estimates over a subset of the $f - \alpha$ plane of interest, rather than the entire plane. The resulting reduction in computational complexity is eminently suitable for cognitive radios.

The question is then – what subset of the $f - \alpha$ plane should the SCD be estimated for? It makes sense to concentrate only on the regions where the SCD is significantly

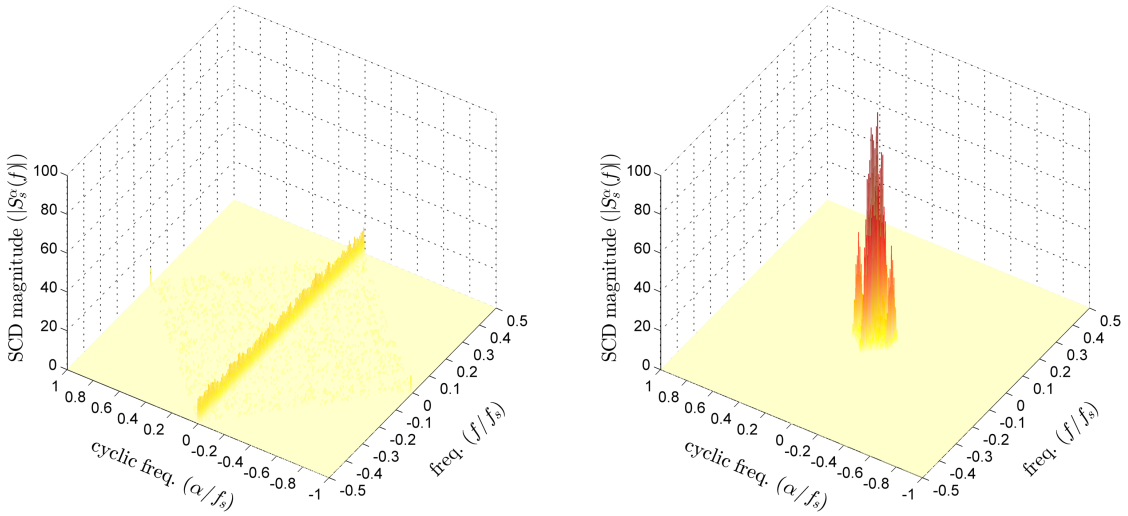

Figure 2.8: An estimate of the SCD (magnitude only) using FAM for: (left) AWGN; (right) QPSK signal with SRRC pulse shaping, 8 samples per symbol, and an observation length of 8192 samples. Both the AWGN and the SRRC signals have the same power.

different between the scenarios when H_0 and H_1 is true for the signal of interest. Referring back to (2.6), any difference between the SCDs for the two scenarios is due to $s_t[n]$ being absent/present.

Therefore, a region Ω is defined as illustrated in Figure 2.9. It corresponds with where one would expect to see a most pronounced change in the SCD between the H_0 and H_1 scenarios.

Therefore, a detection test statistic can be derived that is based on the cyclic spectral energy present in Ω:

$$\lambda = \sum_{(f,\alpha)\in\Omega} |S_s^\alpha(f)|^2, \tag{2.38}$$

where (f, α) is an ordered pair on the $f-\alpha$ plane. By comparing λ to a threshold, λ_T, the hypothesis test is now in the same form as (2.7).

It is noteworthy that, in all of the above discussion, only the top half of the $f-\alpha$ plane has been considered (i.e. when $\alpha \geq 0$). By symmetry, the bottom half of the plane is redundant for the purpose of signal detection and is therefore neglected. This observation also has the favourable side-effect of reducing the computational complexity of the detector even further by a factor of two.

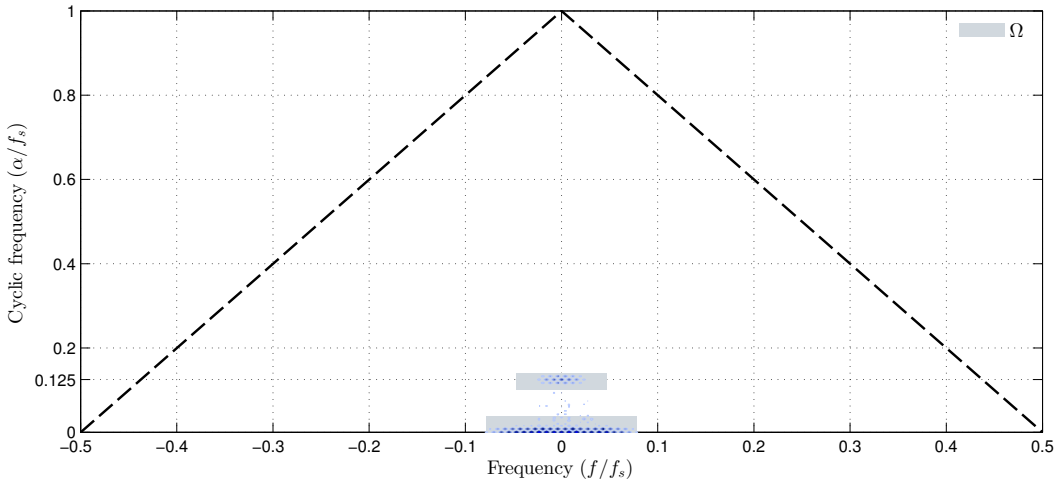

FIGURE 2.9: Region of interest, Ω, for cyclostationarity detection of a SRRC signal with QPSK modulation and 8 samples per symbol, based on its SCD estimate.

2.5.2 Detection Performance

The detection-error performance of a detector based on (2.38) was determined through MATLAB simulations, and compared to:

- A *simulated* cyclostationarity detector employing the full SCD estimate in its test statistic (e.g. as indicated in [56]). In other words:

$$\lambda = \sum_{\forall f,\alpha} \left| S_s^{\alpha}(f) \right|^2 . \tag{2.39}$$

- The theoretical performance expected for an energy detector.

For this purpose, it was assumed that the only channel perturbation was in the form of AWGN. It is assumed that the licensed network user's transmission to be detected has modulation parameters which are known in advance, and following the parameters described in the caption in FIGURE 2.8.

FIGURE 2.10 and FIGURE 2.11 shows detector performance for SNR = -6 dB and $N = 256$ samples, and SNR = -10 dB and $N = 512$. One million detection events were simulated at each detection threshold value for the reduced-complexity detector, whereas 100000 detection events were similarly used for the detector employing the full SCD estimate.

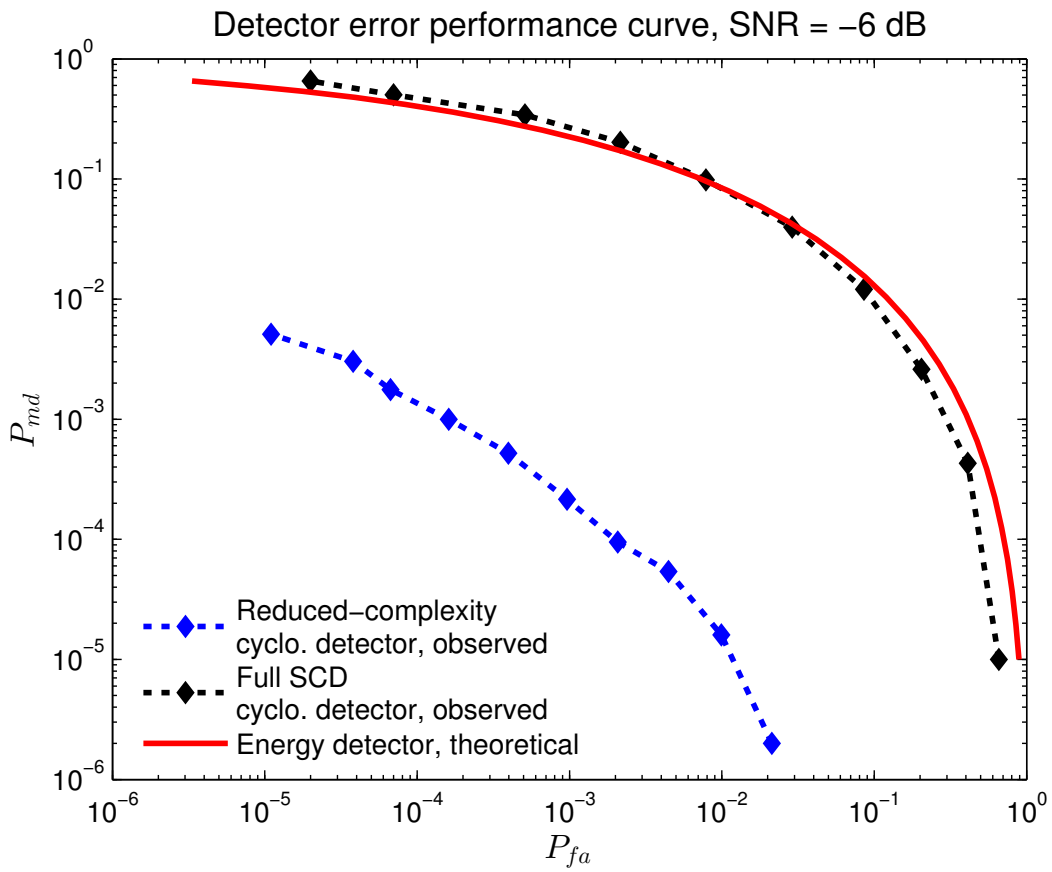

Figure 2.10: Comparison of detection-error performance between a reduced-complexity cyclostationarity detector, a cyclostationarity detector employing the full SCD estimate and an energy detector; $N = 256$, SNR = -6 dB.

For an energy detector, the theoretical probability of false alarm (P_{fa}) is given by [24]:

$$P_{fa} = Q\left(\frac{\lambda_T}{\sqrt{N}\sigma_n^2} - \sqrt{N}\right), \tag{2.40}$$

where σ_Z^2 is the complex noise power (variance), and $Q(x)$ is the Q-function, defined as:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{\frac{-u^2}{2}} du, \tag{2.41}$$

whilst the theoretical probability of missed detection (P_{md}) is given by [24]:

$$P_{md} = Q\left(\frac{\sqrt{N}(\rho+1) - \frac{\lambda_T}{\sqrt{N}\sigma_n^2}}{\sqrt{(\zeta-1)\rho^2 + 2\rho + 1}}\right), \tag{2.42}$$

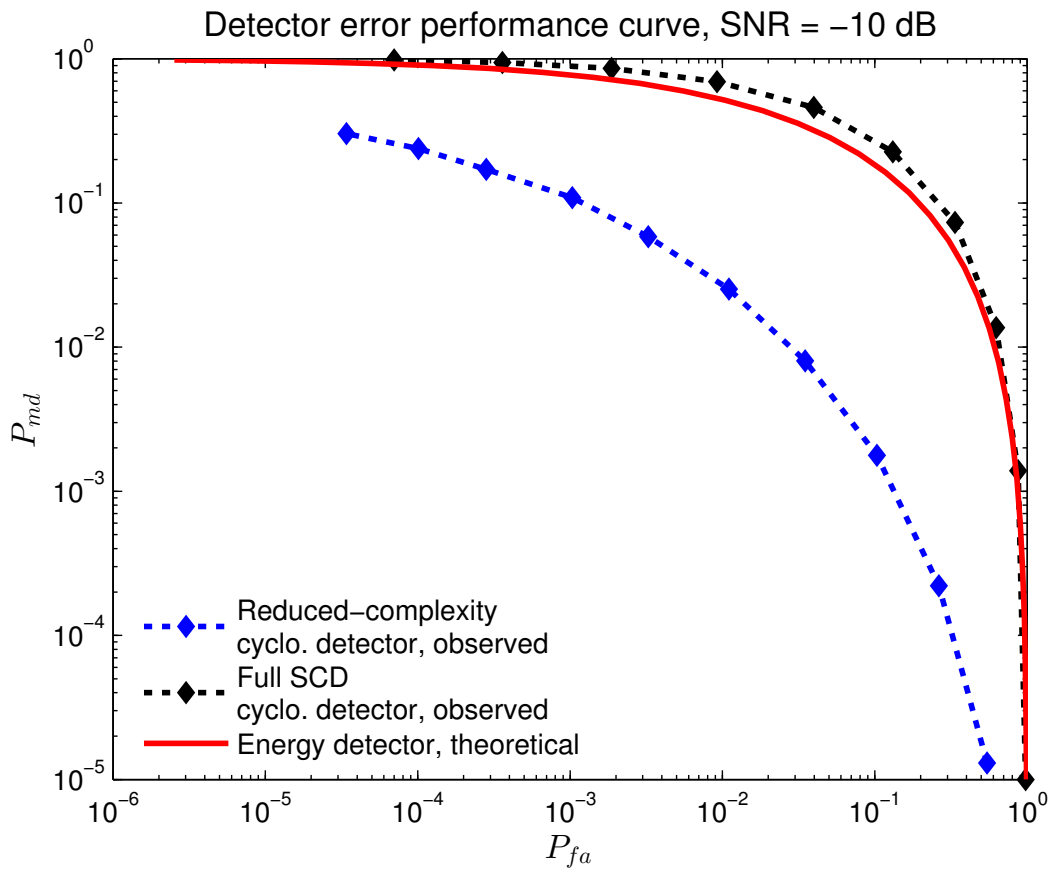

FIGURE 2.11: Comparison of detection-error performance between a reduced-complexity cyclostationarity detector, a cyclostationarity detector employing the full SCD estimate and an energy detector; $N = 512$, SNR = -10 dB.

where ρ is the SNR as defined in (2.13), and $\zeta = \dfrac{E\{|x[n]|^4\}}{E^2\{|x[n]|^2\}}$ is a parameter characterizing the shape of the pdf of the complex envelope magnitude of the signal [24]. $E\{\bullet\}$ is the expectation operator.

In both cases, the reduced-complexity detector has superior detection performance compared to an energy detector. It is also noteworthy that the detection performance of the cyclostationarity detector employing the full SCD estimate is at a level comparable to a simple energy detector. Some concluding remarks about this are made in section 2.5.4.

2.5.3 Energy-Consumption Comparison of Signal Detection Approaches

We present a comparison of the indicative energy consumption of the energy detector, a cyclostationarity detector employing a full FAM-based estimate to the SCD, and the cyclostationarity detector described here.

In order to do this, a reasonable measure of the comparative computational complexity of these detection schemes needs to be established – in this case, the metric used is the number of complex multiply operations required per detection operation. The detection operation consists of two conceptual parts:

- Analysis of a signal observation of length N samples
- Calculation of the test statistic from the analysis results

In using the number of multiply operations as a measure of computational complexity, it is assumed that multiplication is the dominant consumer of processor clock cycles in a software implementation, allowing other arithmetic operations such as additions to be neglected. Processor cycles and pipeline stalls relating to other operations, such as memory accesses, are also neglected.

Energy Detector

Based on (2.11), the number of complex multiplies needed to calculate the test statistic is given by:

$$C = N, \tag{2.43}$$

since the magnitude squared of a complex signal sample is equal to the sample value multiplied by its conjugate.

Full Frequency-Domain Cyclostationarity Detector

For such a detector, it is assumed that it will obtain an SCD estimate for the *entire* $f - \alpha$ plane, using FAM. The following are the relevant FAM parameters which have

been used (described in more detail in [70] and [64]):

- Time-frequency resolution product of $M = \dfrac{\Delta f}{\Delta \alpha} = 2$

It can be shown that the number of complex multiplies needed to obtain a test statistic is of the order:

$$C = \underbrace{\frac{NN'}{L}\left(\log_2 N' + N'\left(1 + \log_2\left(\frac{2N'}{L}\right)\right)\right)}_{C_S} + \underbrace{\frac{2N'^3}{L}}_{C_T}. \tag{2.44}$$

A justification of (2.44) is provided in section A.4.

C has two components. The first component, C_S, is the complex multiplies needed to obtain SCD estimates within each of the N'^2 CPRs, whilst making use of all N samples in the signal observation. The second component, C_T, is the complex multiplies needed to calculate the test statistic by summing every SCD estimate that was obtained across the entire $f - \alpha$ plane – again, assuming that the magnitude squared of each SCD estimate can be obtained by multiplying it by its conjugate.

Reduced-Complexity Frequency Domain Cyclostationarity Detector

For the reduced-complexity detector described, FAM is also used, but only evaluated for ordered pairs satisfying $(f, \alpha) \in \Omega$. In this specific formulation, it can be shown that the number of complex multiplies to obtain the test statistic is of the order (refer to section A.5 for details):

$$C = \underbrace{\frac{NN'}{L}\left(\log_2 N' + \frac{3}{256}N'\left(1 + \log_2\left(\frac{2N'}{L}\right)\right)\right)}_{C_S} + \underbrace{\frac{3N'^3}{128L}}_{C_T}. \tag{2.45}$$

Again, C has two components. C_S is the complex multiplies needed to obtain SCD estimates over Ω, resulting in C_T SCD value estimates – C_T complex multiplies then follow to obtain the test statistic.

	Scorpion	Krait
Benchmark, DMIPS/MHz/core	2.1	3.3
Typical clock rate, GHz	1.0	1.5
Total DMIPS/core	2100	4950
Typical power usage, mW/core	350	265
Energy efficiency, DMIPS/mW	6.0	18.7

Table 2.2: Comparison of some CPUs found in existing mobile devices. DMIPS stands for *Dhrystone MIPS* – million instructions per second according to the Dhrystone benchmark.

Comparison of Energy Consumption

From the computational complexity values, it is possible to then use an embedded processor's power usage to estimate how the detection schemes compare in terms of energy efficiency – particularly, the energy consumed per detection operation. For this exercise, the following two processors (CPUs) from Qualcomm are compared:

- Scorpion, based on ARM's Cortex-A8 and -A9 processor architectures, and incorporated into the Snapdragon S1, S2 and S3 mobile SoCs
- Krait, based on ARM's Cortex-A15 processor architecture, and incorporated into the Snapdragon S4, 400, 600 and 800 mobile SoCs

Some published specifications for both CPUs are illustrated in Table 2.2 – the information has been acquired from several sources [73], [74], [75], [76], [77].

Given that one complex multiply operation is equal to four real multiplies and two real additions, and if it is assumed that both processors can achieve this using one single-instruction-multiple-data (SIMD) multiply and one SIMD addition instruction, and that this is comparable to two Dhrystone 'instructions' then, using the DMIPS/mW figure from Table 2.2, the energy expended per detection operation can be estimated for each detection scheme.

Based on this model, Figure 2.12 shows the relative energy usage of the three aforementioned detectors, for various N, whilst $N' = 128$. For the detector based on a full SCD estimate, it was also assumed that no decimation was applied (such that $L = 1$). On the other hand, for the low-complexity cyclostationarity detector,

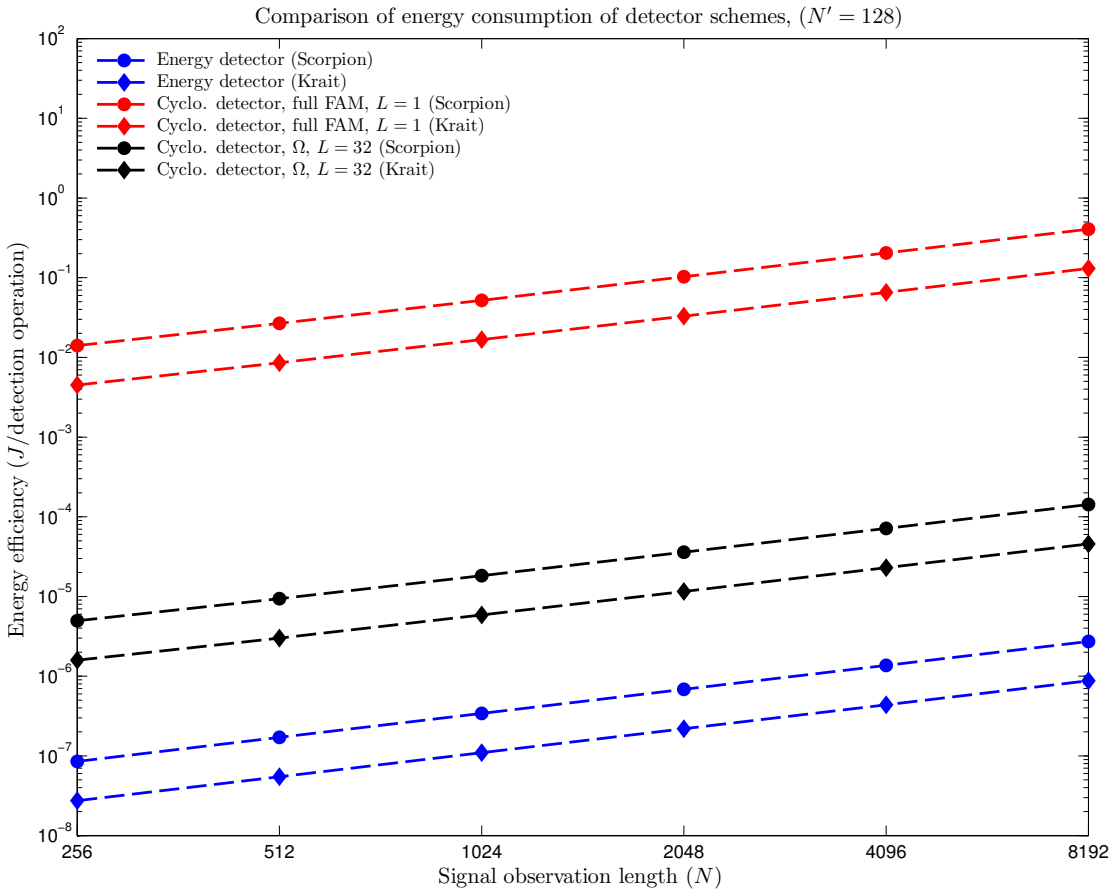

FIGURE 2.12: Comparative efficiency of energy and cyclostationarity signal detection approaches.

it was assumed that a decimation factor of $L = \frac{N'}{4} = 32$ was used. It can be seen that the energy consumption of cyclostationarity detection can be significantly reduced compared to a full-SCD implementation.

2.5.4 Remarks: Why Employing the Complete Spectral Correlation Density Estimate Is Not a Good Idea

FIGURE 2.10 and FIGURE 2.11 indicated that when a the cyclostationarity detector computed its test statistic using the entire SCD estimate, the resulting detection performance was comparable to a simple energy detector. The extra computation expended did not yield any improvement in detection performance!

Points on the f, α plane fall into one of the following categories:

- *Feature points*, whose value is obtained from appreciably different pdfs depending on whether H_0 or H_1 is true;
- *Non-feature points*, whose value is obtained from the same pdf regardless of whether H_0 or H_1 is true.

The poor performance arises from the fact that in most cases including this one, the number of non-feature points points on the f, α plane is much larger than the number of feature points. Therefore, including them in the test statistic as indicated in (2.39) actually dilutes its ability to discriminate between the H_0 and H_1 cases. More formally, it causes the mean value of the test statistic when H_1 is true to approach that when H_0 is true – bringing the two pdfs closer together results in poorer detection performance as indicated in Figure 2.2.

This observation demonstrates the *need* to identify which points on the f, α plane represent genuine features that provide useful information to differentiate between signals representative of the presence and absence of the licensed network user's transmission on a channel. This is where additional computational resources should be employed, and is the concern of the remainder of this chapter.

2.6 Cyclostationarity Detector: Exploiting Computation Instead of Prior Knowledge

In the preceding section, it was assumed that it was known, in advance, what regions of the $f - \alpha$ plane changed the most markedly between the H_0 and H_1 scenarios. The detector can then concentrate on these regions in a bid to reduce the computational effort associated with spectrum sensing in operation.

However, different signals will have different cyclostationarity features; a detector that only relies on prior knowledge is fundamentally inflexible. A mobile cognitive radio device may move between different radio environments, and so require detectors that can adapt in the field by having some ability to train themselves to locate these dominant cyclostationarity features. Here, it is demonstrated that *genetic algorithms*

(GA) can be used to search out the locations of these features in a fairly arbitrary way – thus making them suitable for detecting a multitude of signal transmission types.

2.6.1 Extracting Dominant Cyclic Spectral Features as Part of Adaptive Training

In terms of the GA terminology described by [78], the SCD can be considered a 'chromosome', defined by two genes representing the ordered pair (f, α). GAs can hence be used to locate dominant features (i.e. *regions of maxima*) associated with a signal's SCD magnitude.

In general, it has been observed that the points of absolute maxima in the SCD tend to lie along the PSD line, so a GA will tend to gravitate towards these points preferentially. However, it is desirable to exploit features that might exist across the entire $f - \alpha$ plane, making use of as much information about the signal as is available. (Otherwise, as has been discussed in section 2.4, the detection-error performance converges towards that of energy detection.) To this end, the $f - \alpha$ plane is compartmentalized into two regions, Φ_0 and Φ_1, with separate GA searches carried out on each. FIGURE 2.13 illustrates these two regions. The SCD of a QPSK signal has been superimposed for illustration. Again, symmetry is exploited, and both computation and the GA search area is reduced, by only considering the top half of the plane ($\alpha \geq 0$). Φ_0 encapsulates the PSD values.

The search regions Φ_0 and Φ_1 can be defined as:

$$\begin{aligned} \Phi_0 &= (\alpha \leq 2f + f_s) \cap (\alpha \leq -2f + f_s) \cap \\ &\qquad (\alpha < b) \cap (\alpha >= 0) \\ \Phi_1 &= (\alpha \leq 2f + f_s) \cap (\alpha \leq -2f + f_s) \cap (\alpha \geq b). \end{aligned} \tag{2.46}$$

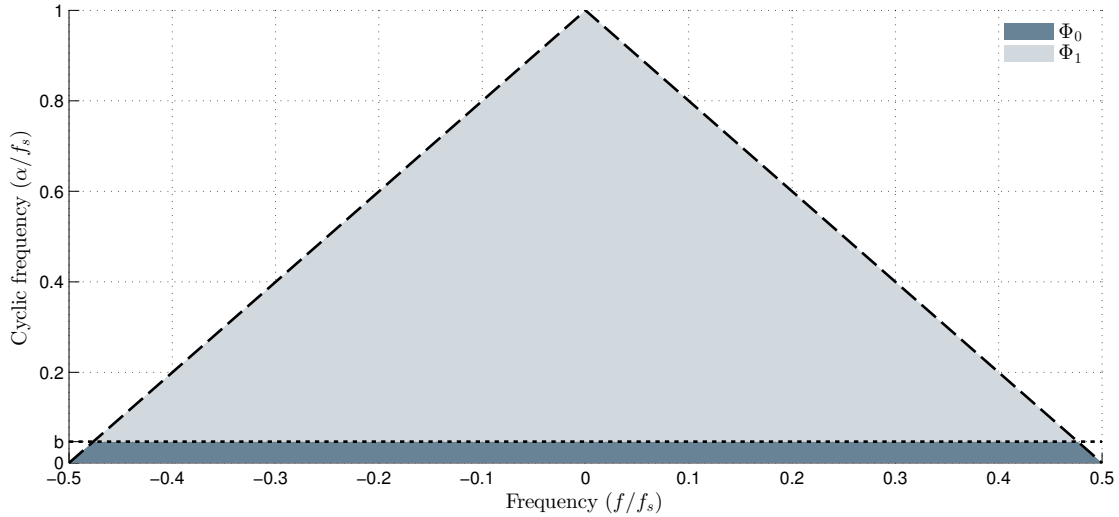

Figure 2.13: Division of the $f-\alpha$ plane into Φ_0 and Φ_1 regions.

In principle, we use two GAs to perform the following optimizations[4]:

$$\arg\max_{\Phi_0} |S_s^\alpha(f)|^2, \qquad \arg\max_{\Phi_1} |S_s^\alpha(f)|^2. \tag{2.47}$$

Extracting Spectral Features from a Single Training Observation

The optimization in (2.47), strictly speaking, will yield an estimate of the *absolute* maxima of the SCD estimate in each region, whereas the dominant features of the cyclic spectrum generally consist of a set of points representing *relative* maxima – i.e. the G points with the highest value, where G is a positive integer. It would be desirable to run the GA optimization multiple times and obtain unique points each time, hence constructing a 'picture' of these dominant features. To ensure that this happens, the GA operates on a sequence of *training observations*, where *it is known in advance that the signal of interest is present*, according to a procedure like this applied to Φ_0:

- Let the search area at iteration g be Φ_0^g, where $g = 1, 2, \ldots, G_0$ and G_0 is a positive integer.

[4]The 'unsquared' magnitude could also be used - although it is imagined for most processor instruction sets that the squared magnitude could be obtained with fewer instructions.

- At the first iteration, $g = 1$, the search area, Φ_0^1, is defined as in (2.46):

$$\Phi_0^1 = \Phi_0. \tag{2.48}$$

- At iteration g, the GA estimates the following:

$$(f_{\Phi_0}^g, \alpha_{\Phi_0}^g) = \arg\max_{\Phi_0^g} |S_s^\alpha(f)|^2. \tag{2.49}$$

- A small rectangular area, δ_g, is defined whose centre is the point $(f_{\Phi_0}^g, \alpha_{\Phi_0}^g)$. This area is then 'subtracted out' from Φ_0^g for the next iteration, thus preventing subsequent iterations of the GA from locating a point in the immediate vicinity of that just obtained:

$$\Phi_0^{g+1} = \Phi_0^g - \delta_g. \tag{2.50}$$

- After G_0 iterations, $(f_{\Phi_0}^1, \alpha_{\Phi_0}^1), (f_{\Phi_0}^2, \alpha_{\Phi_0}^2), \ldots, (f_{\Phi_0}^{G_0}, \alpha_{\Phi_0}^{G_0})$ is a sequence of *distinct* points representing an estimate of the dominant SCD features for the current signal observation, in Φ_0.

The above description applies for Φ_0, but is applied similarly for Φ_1, to obtain a sequence of G_1 points $(f_{\Phi_1}^1, \alpha_{\Phi_1}^1), (f_{\Phi_1}^2, \alpha_{\Phi_1}^2), \ldots, (f_{\Phi_1}^{G_1}, \alpha_{\Phi_1}^{G_1})$ in Φ_1.

Combining Spectral Features from Multiple Training Observations

It is necessary to train the detector over a *series* of observations, to smooth out aberrations that may be present in a single, relatively short observation. Multiple observations ensure that information is extracted from the signal over a reasonable timeframe.

From the points $(f_{\Phi_0}^1, \alpha_{\Phi_0}^1), (f_{\Phi_0}^2, \alpha_{\Phi_0}^2), \ldots, (f_{\Phi_0}^{G_0}, \alpha_{\Phi_0}^{G_0})$ obtained from all training observations, the K_0 points that appear most often across all the observations are selected – this forms the set of points ϕ_0. Similarly, the set of points ϕ_1 from the K_1 points that appear most often across $(f_{\Phi_1}^1, \alpha_{\Phi_1}^1), (f_{\Phi_1}^2, \alpha_{\Phi_1}^2), \ldots, (f_{\Phi_1}^{G_1}, \alpha_{\Phi_1}^{G_1})$, are formed. The final

Parameter	Description
Chromosome population size	32
Stopping condition for a GA run	32 generations
Elite count (chromosomes selected for next generation)	4
Crossover/mutation fraction (of non-elite chromosomes for next generation)	0.5/0.5
Mutation function	Add Gaussian random value, 'clip' to FAM plane boundaries
Boundary between Φ_0 and Φ_1 (b)	$\frac{6}{128}$
No. training observations	32

Table 2.3: Genetic algorithm simulation parameters.

set of points, Ω, used by the detector after training, is defined as:

$$\Omega = \phi_0 \cup \phi_1. \tag{2.51}$$

The detection test statistic is then defined, for any signal observation during spectrum the sensing operation, as:

$$\lambda = \sum_{\Omega} |S_s^{\alpha}(f)|^2. \tag{2.52}$$

2.6.2 Adaptation and Detection Performance

In this section, MATLAB simulation results are presented to show both detector adaptability and resulting detection performance, once (2.52) is applied. For these simulations, *it is assumed that the GA only runs whilst the licensed network user is transmitting*. In other words, the GA *never* trains itself on a signal observation consisting of only noise. (This actually cannot happen in practice – a point that is elaborated upon later in Chapter 3.) Some key GA parameters used are described in Table 2.3.

In the first instance, the same licensed network user modulation characteristics were used as for the simulation with SNR = -6 dB and $N = 256$ in section 2.5.2. The same FAM calculation parameters were also employed.

Figure 2.14 shows the set of points, Ω, selected by the GA for use in the detection

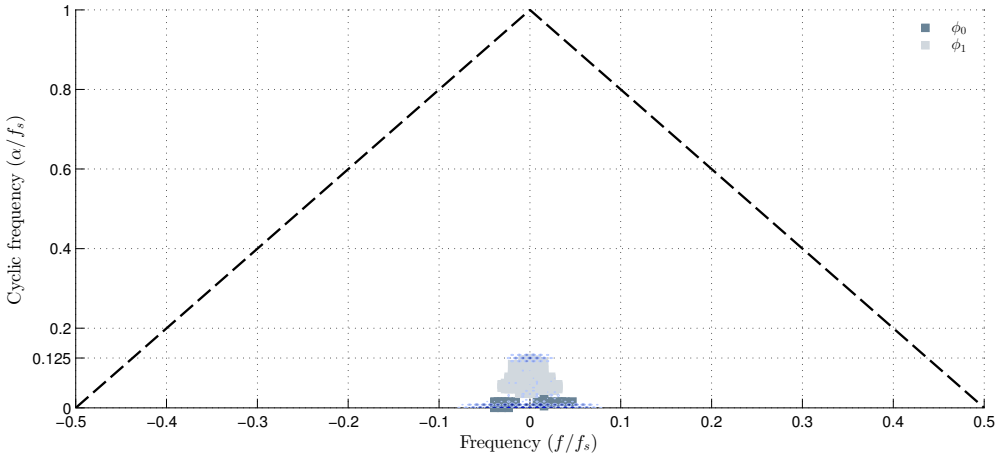

FIGURE 2.14: Regions of interest identified by a genetic algorithm for a SRRC signal with QPSK modulation and 8 samples per symbol.

statistic; $K_0 = 64$ and $K_1 = 192$. Superimposed is an example of an SCD contour plot expected for sufficiently long observations of the signal of interest. FIGURE 2.15 shows the performance of the detector in simulation – once again its performance is superior to that expected of an energy detector.

To demonstrate the GA-based adaptation, the simulation was then re-run for the scenario of detecting a licensed network user employing QPSK symbols over OFDM. Each OFDM symbol had a length of 1152 samples (1024 samples, with a guard interval of $\frac{1}{8}$), with 8 equally-spaced, unmodulated pilot carriers of amplitude 12. A greater observation length of $N = 512$ samples was used. FIGURE 2.16 shows the expected SCD features of this signal given a sufficiently long observation; the 'speckles' are due to the presence of the equally-spaced pilot carriers. It also shows the set of points, Ω, discovered by the GA for this scenario; $K_0 = 192$ and $K_1 = 384$. The 'speckles' are captured by ϕ_1. FIGURE 2.17 shows that the resulting detector's performance is again superior to energy detection (for the same N).

The additional computation expended in extracting cyclic spectral features from a signal of interest from a GA-based training process, in lieu of prior knowledge about the signal of interest, yielded a superior detector compared to energy detection.

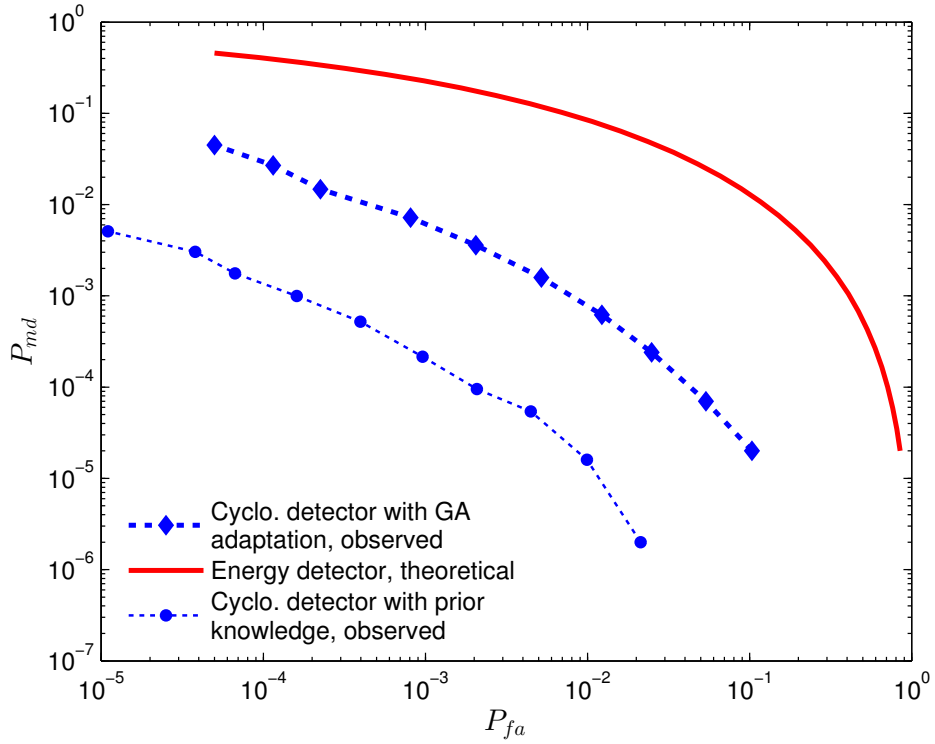

FIGURE 2.15: Cyclostationarity detector performance after GA adaptation and comparison with energy detection; SRRC pulse shaping, QPSK modulation, $N = 256$, SNR = -6 dB.

2.7 Concluding Remarks

In this chapter, the fundamental aim was to demonstrate a cyclostationarity detector that was able to operate effectively without having to calculate an entire SCD or CAF estimate of a signal observation. To do this, the detector focuses its computational effort only on the areas of the SCD that change most pronouncedly, between the cases of H_0 and H_1 being true. The reduction in computational complexity is achieved by concentrating on these regions, identified by exploiting knowledge about the licensed network user's signal. Associated with this is an energy efficiency gain, making cyclostationarity detection suitable for power-limited devices.

The first version of this detector achieves this gain by exploiting prior knowledge about the signal being detected. The second version uses additional computation in a training process using GAs to learn where on the $f - \alpha$ plane the detector should be

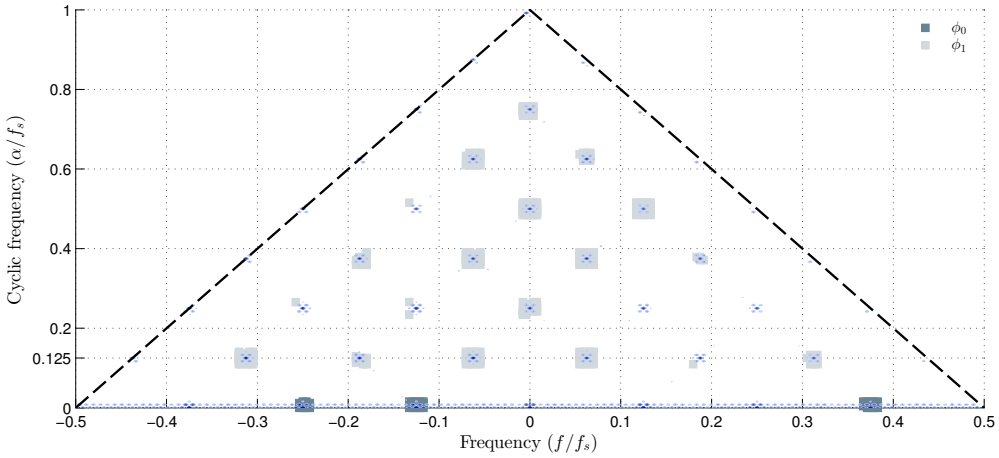

FIGURE 2.16: Regions of interest identified by a genetic algorithm for an OFDM signal with 1024 subcarriers per symbol, of which 8 are pilot subcarriers spaced evenly apart, and the rest are modulated with QPSK.

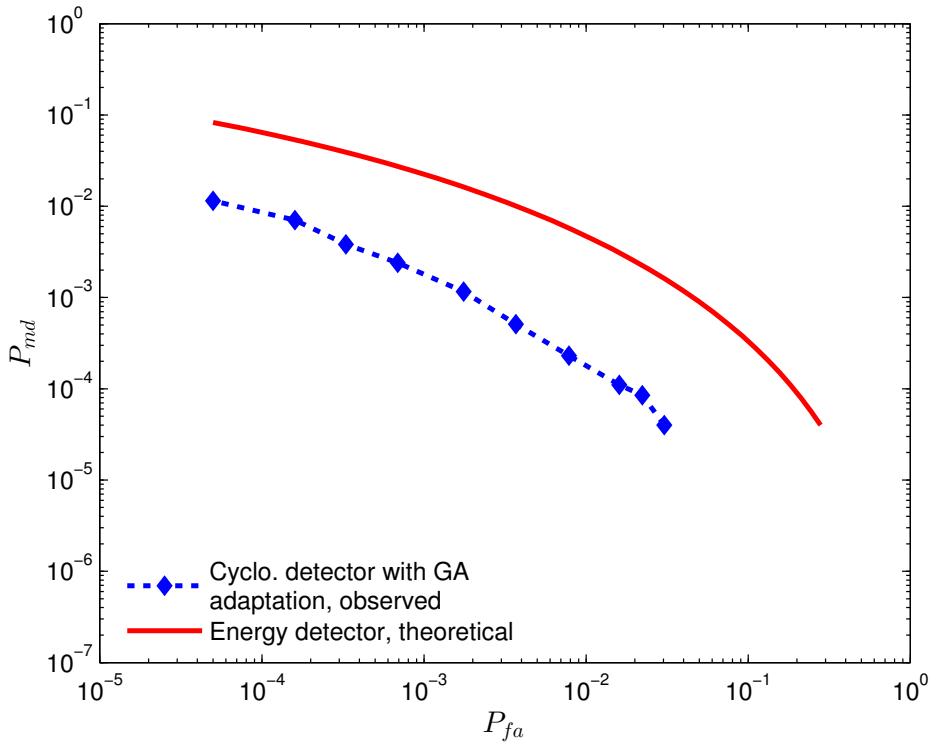

FIGURE 2.17: Cyclostationarity detector performance after genetic algorithm adaptation and comparison with energy detection; QPSK symbols over OFDM, $N = 512$, SNR = -6 dB.

concentrating – such a detector is adaptable and able to work effectively even as the cyclostationarity characteristics of the licensed network user's transmission changes.

As an aside, it may be possible to determine the cyclostationarity properties of

a licensed network user by simply performing a sorting operation on the SCD point estimates obtained. However, the number of values to be sorted may be large, resulting in a highly computational-intensive implementation.

3

Learn: Signal Detectors that Self-Configure in the Field

In the previous chapter, the underlying principles of signal detection techniques exploiting the signal property of cyclostationarity were presented, including a discussion of how both the cyclic autocorrelation function and the spectral correlation density could be exploited. The topic was explored in the context of several highly desirable traits of cognitive radio devices performing spectrum sensing for dynamic spectrum access – namely, low detection error rate (specifically, better than energy detection), reduced computational complexity (and hence reduced-power consumption in a software-defined radio implementation), and adaptability to different radio environments.

Adaptability in this context refers to the ability for a detector to perform its job, regardless of whatever modulation scheme or parameters happen to be utilized by the

licensed network user operating in its vicinity. In particular, mobile wireless devices can expect to be subjected to dynamic, rapidly changing radio environments as they move between different locales. Therefore, it is not realistic to expect that a cognitive radio be statically pre-programmed with all the information required to detect all classes of licensed network devices in the field. (In fact, such a cognitive radio would not be very 'cognitive' at all.)

The groundwork for exploiting cyclostationarity in a *feature-based* detector was discussed in the previous chapter. Particularly, the desirable characteristic of adaptability was demonstrated through 'training' the detector. Using a representative suite of signal observations of a licensed network user's transmission (a *training set*), the locations of the dominant discriminating features in its SCD on the $f - \alpha$ plane were identified. It was assumed, perhaps tacitly, that the detector has a 'perfect' training set with which to work – meaning that all signal observations used actually contain the transmission of interest.

However, for a device to train itself in the field, it needs to construct its training set from signal observations taken from its radio environment – and therefore this assumption cannot be strictly true. The ability to perfectly discriminate between signal observations where the licensed network user's transmission is present and absent, for the purposes of obtaining a training set, implies that a perfect signal detector already exists!

Therefore, in this chapter:

- The concept of *imperfect training* as it relates to cognitive radios with adaptive signal detection schemes operating in the field is introduced
- A general framework for training feature-based detectors, such as cyclostationarity detectors, in the field, is presented, based on a two-stage detector implementation fulfilling both a *training* and *operational* requirement
- The deteriorative effects of imperfect training are described
- A relationship between time spent on training, and the amount of uncertainty

associated with whether or not observations within the training set are truly representative of the signal of interest, is developed

- An empirical investigation into the effect of this training-set uncertainty on an adaptive cyclostationarity detector's detection error performance is presented

3.1 Training a Feature-Based Signal Detector in the Field

3.1.1 The Imperfect Teacher Problem in Context

In section 2.6.1, a cyclostationarity detector is trained using a genetic algorithm to discover the regions on the $f - \alpha$ plane where an SCD estimate is likely to yield a relatively high magnitude, if a licensed network user is transmitting during the observation window from which the SCD estimate is made. Collectively, the points in these regions make up the SCD *features* of the signal of interest.

To perform this training, a series of unique observations is used, which can be considered to be a training set in a *supervised learning* [79] exercise. The cognitive radio, given a training set of signal observations (as an input), attempts to learn a set of detection model parameters (points on the $f - \alpha$ plane) such that, when a function is applied on the input using these parameters (a summation of the SCD magnitude estimates over these points and compared to a detection threshold), a correct detection result (the output) occurs. These signal observation 'samples' which make up the training set are assumed to be correctly '*labelled*' – in other words, the 'teacher' that is conducting the supervised learning is *perfect* and never mislabels any observations that are used for training. The terms 'teacher' and 'label' are used in works such as [80].

An example may serve to illustrate this point further. As binary signal detectors are being considered, any observation used in the training exercise can be labelled in one of two ways (similarly to how a detection result on an observation can have one of two outcomes, as shown in (2.7)):

$$L \in \{0, 1\},$$

$$L = \begin{cases} 0 & \text{if the observation is deemed unsuitable for training} \\ 1 & \text{if the observation is deemed suitable for training,} \end{cases} \tag{3.1}$$

where L is the label attached by the teacher to a signal observation. By definition, only observations where $L = 1$ are used for supervised learning.

Each observation also has an underlying 'reality' of whether or not the signal of interest is, in fact, absent or present, as described by (2.2). Therefore, an observation is *mislabelled* when:

$$L \neq R. \tag{3.2}$$

In a perfect training environment, assumed in section 2.6.1, there is never any mislabelling. Under imperfect training, however, there is some probability that observations used within the training set have been mislabelled [80]:

$$L = 1 \text{ but } R = 0. \tag{3.3}$$

This can be considered a *false inclusion* into the training set.

If a cognitive radio undergoes self-training whilst in the field, it is not possible to have a perfect training environment. This is because the training set must come from observations that it takes from its radio environment. How does it know, when it takes an observation, how to label it? A decision must be made as to what label to apply; *if it was possible to guarantee correct labelling, then this implies that a perfect signal detector already exists!* This is impossible, so it is concluded that, if a cognitive radio undergoes adaptation when operating in the field in response to some form of learning process, this will necessarily be driven by an imperfect training environment.

3.1.2 A Model for Imperfect Training of a Feature Detector Using Field Observations

To initiate a training process on a feature detector, firstly the teacher must collect a sequence of observations from the radio environment, labelled with $L = 1$, to form the training set.

The labelling must be driven by a decision arrived at for each signal observation taken – this is exactly the same detection problem portrayed in (2.7)! Consequently, in order to carry out this labelling process, the concept of a *screening detector* is introduced. It employs the same decision process as in (2.7), *with the label being equal to the decision.* In other words, for each candidate training observation, based on the evidence present, the screening detector makes a binary decision about whether the licensed network user's transmission is currently absent (apply label $L = 0$) or present (apply label $L = 1$). If the screening detector decides $L = 1$, then the observation is added to the training set, and discarded otherwise. Once the training set has been collated, learning of the features present within it that could enhance detection performance can take place (e.g. see section 2.6.1), and used within a feature detector for actual operation. It is apparent from this discussion, and referring to (3.3), that *the imperfection in the training comes from the screening detector's probability of false alarm,* P_{fa}.

The two-stage detection model is illustrated in Figure 3.1. Whilst two-stage detectors have been described previously to achieve various ends relating to spectrum sensing, such as coarse/refined detection for improved performance [58] and coarse/fine resolution of occupancy sensing across a wide bandwidth [57], in this work, two stages are used to address the problem of detector *training* followed by *operation.*

The conceptual idea is that a screening detector is used to 'bootstrap' a learning process, which uncovers hidden features within the signal observations used in the training set, which then is used as an input into a feature detector – of course, the end result needs to be that the resulting feature detector has better detection-error performance than that of the screening detector, due to the extra analysis and computation

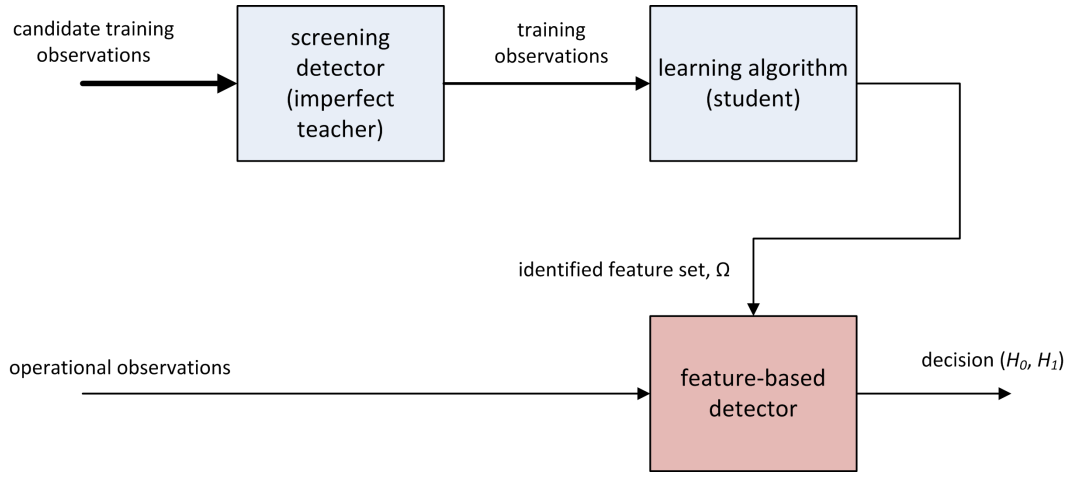

FIGURE 3.1: A model for (imperfect) training of a feature-based detector for cognitive radio, using observations taken from the field that have been labelled by a screening detector.

associated with the learning process.

Note that, up until this point, the screening detector has been described in fairly abstract terms. An ideal screening detector should be computationally efficient and rely on as little information as possible in relation to the features and structures within the signals of interest – the well-studied energy detector meets these criteria, and is used in subsequent work within this chapter. Also, as in the previous chapter, the feature detector is a cyclostationarity detector. Despite these specifics, the model shown in FIGURE 3.1 has no particular reliance on energy or cyclostationarity detectors – in principle, any reasonably performing detection algorithm (i.e. one that performs better than a 'random guess') could be used for the screening detector, for example. The model is also agnostic to the specific details of the learning algorithm (i.e. how the training set is analyzed such that the underlying features may be identified and extracted from the training set).

3.1.3 Deteriorative Effects Due to Imperfect Training

Ideally, a learning algorithm would exhibit all of the following traits:

- Short learning time (as time spent training is an opportunity cost on transmission, and requires energy to be expended for this duration)

- Identify signal features robustly (i.e. as low a detection error rate as possible)

Consider an idealized, perfect training scenario where the screening detector never mislabels observations that are used for training. In comparison to this, imperfect training may result in identifiable degradations to both of the aforementioned outcomes.

Assume that there is a minimum required number of observations which need to be taken in order to perform training (i.e. the size of the training set), N_{ts}. Also, assume that the training observations are taken from a channel where the probability of the licensed network user of interest transmitting is equal to P_1 – therefore, the probability that the user does *not* transmit is given by $P_0 = 1 - P_1$. Therefore, the *average* number of observations that must be taken in order to populate the training set accordingly, in the perfect training scenario, will be:

$$N_{pt} = \frac{N_{ts}}{P_1}. \tag{3.4}$$

If each observation consists of N signal samples taken at a sampling period of T_s, and the average time period between the start of the collection of one candidate observation and the end of the previous observation is T_o, then the average time required to populate the training set, in this perfect training scenario, can be given as:

$$T_{ts} = N_{pt} \left(NT_s + T_o \right). \tag{3.5}$$

However, by considering N, T_s and T_o as constants, $T_{ts} \propto N_{pt}$ – by neglecting the proportionality constant $(NT_s + T_o)$, N_{pt} may be considered to be a measure of the total time required to populate the training set under the perfect training scenario.

Under imperfect training, it is possible that the number of observations that must be made to populate the training set could be greater than N_{pt}. This is because the screening detector, being imperfect, and having some non-zero probability of missed detection (P_{md}), may erroneously exclude from the training set an observation which should be included because the signal of interest is actually present; more formally:

$$L = 0 \text{ but } R = 1. \tag{3.6}$$

This represents a *missed training opportunity* - or a *false exclusion*.

It is important to recognize that the detection-error performance of the feature detection, once training has been completed, is also affected by imperfect training, in the sense that *the features identified by the training algorithm may not be optimal*. This is because the screening detector, being imperfect, and having some P_{fa}, may incorrectly admit observations into the training set due to the incorrect labelling in (3.3). This may 'dilute' the information about the features present in the licensed network user's transmission in the training set as a whole, which in turn makes it more difficult for the learning algorithm to identify these features, and subsequently degrade the operation of the feature-based detector in its operational phase.

3.2 Analysis of Required Training Times Under Imperfect Training

It has been shown that all practical signal detectors are able to trade off P_{fa} and P_{md} by varying the detection threshold used. Suppose that the screening detector calculates a test statistic λ_{sc} from a candidate training observation, and uses a detection threshold $\lambda_{T,sc}$ to determine how it labels such observations – this results in (3.1) being expressed more specifically as follows:

$$\begin{aligned} &L \in \{0,1\}, \\ &L = \begin{cases} 0 & \text{if } \lambda_{sc} < \lambda_{T,sc} \\ 1 & \text{if } \lambda_{sc} \geq \lambda_{T,sc}. \end{cases} \end{aligned} \tag{3.7}$$

As a result, under imperfect training, the number of observations that must be taken in order to obtain N_{ts} observations with label $L = 1$ to populate the training set is, on average:

$$\begin{aligned} N_{it} &= \frac{N_{ts}}{P\left(\lambda_{sc} \geq \lambda_{T,sc}\right)} \\ &= \frac{N_{ts}}{P_0 P\left(\lambda_{sc} \geq \lambda_{T,sc} | R=0\right) + P_1 P\left(\lambda_{sc} \geq \lambda_{T,sc} | R=1\right)} \\ &= \frac{N_{ts}}{P_0 P_{fa,sc} + P_1\left(1 - P_{md,sc}\right)}, \end{aligned} \tag{3.8}$$

where R is defined as per (2.2), and the screening detector's probability of missed detection and false alarm are denoted $P_{md,sc}$ and $P_{fa,sc}$ respectively. $P\left(E\right)$ denotes the probability of an event E occurring.

By taking a ratio between (3.8) and (3.4), a *training time degradation factor*, D, can be defined:

$$D = \frac{N_{it}}{N_{pt}} = \frac{P_1}{P_0 P_{fa,sc} + P_1\left(1 - P_{md,sc}\right)} = \frac{P_1}{\left(1 - P_1\right) P_{fa,sc} + P_1\left(1 - P_{md,sc}\right)}. \tag{3.9}$$

If $D > 1$, then the training time has been degraded (is longer) due to the imperfect screening detector, compared to the perfect training scenario; conversely, if $D < 1$, the training time has actually been improved.

3.2.1 Screening Detector based on Energy Detection

Where an energy detector is used as the screening detector, its probabilities of false alarm and missed detection have been well-defined, as it is known that the test statistic follows a χ^2-distribution (see section 2.2.1), allowing (3.9) to be evaluated for this case.

3.2.2 Asymptotic Behaviour for Increasing SNR

Suppose that $\lambda_{T,sc}$ is set to meet a specific value of $P_{fa,sc}$. (i.e. the tolerable rate of false inclusions during training is specified). For high SNR, defined in (2.13) as ρ, the mean values of λ_{sc} for the distinct cases of $R = 0$ and $R = 1$ will be disparate. (i.e. the pdfs of λ_{sc} are 'spread far apart' - see Figure 2.2.)

In such a case, for a given $\lambda_{T,sc}$, as $\rho \to \infty$, $P_{md,sc} \to 0$, allowing an asymptotic approximation to D to be made from (3.9):

$$D \to \frac{P_1}{(1-P_1)\,P_{fa,sc} + P_1} \text{ as } \rho \to \infty. \tag{3.10}$$

3.2.3 Visualization and Verification

FIGURE 3.2 shows a plot of the training time degradation factor, D, against $P_{fa,sc}$ for various low SNR scenarios, as predicted by (3.9), with $P_0 = P_1 = 0.5$. For very low SNR values (-9, -7.5, -6 dB in this case), $D > 1$ for low $P_{fa,sc}$ (i.e. high $\lambda_{T,sc}$). Whilst a high value of $\lambda_{T,sc}$ tends to prevent many false inclusions (which is desirable), it also causes many false exclusions (which is undesirable). Since at low SNR the pdfs $f(\lambda_{sc}|R=0)$ and $f(\lambda_{sc}|R=1)$ are relatively close together, and $P_0 = P_1$, the missed training opportunities become dominant for low $P_{fa,sc}$ (high $\lambda_{T,sc}$), so $D > 1$. As $P_{fa,sc}$ increases ($\lambda_{T,sc}$ decreases), the erroneous inclusions into the training set start to dominate, such that $D < 1$.

FIGURE 3.3 shows the same information, but for different values of P_1, and SNR is fixed at -3 dB. As P_1 decreases, false exclusions have less effect on the training time, causing D to decrease more rapidly as a function of $P_{fa,sc}$.

FIGURE 3.4 shows D for a low-occupancy channel ($P_1 = 0.1$), but also illustrates how the asymptotic expression from (3.10) becomes a better approximation as SNR increases. In this scenario, it becomes a good approximation over the shown range of $P_{fa,sc}$ for SNR > -5 dB.

These results were obtained from MATLAB Monte-Carlo simulations. Some simulation parameters used are shown in TABLE 3.1. Over repeated simulation runs, for a given value of $\lambda_{T,sc}$, the simulation calculated an empirical average for the total number of candidate training observations that needed to be generated (N_{it}) before the N_{ts} observations needed for the training set, labelled as $L = 1$ according to (3.7), were obtained. The observations were generated with the presence of the licensed network user's transmission varying randomly, according to its occupancy statistics defined by (P_0 and P_1).

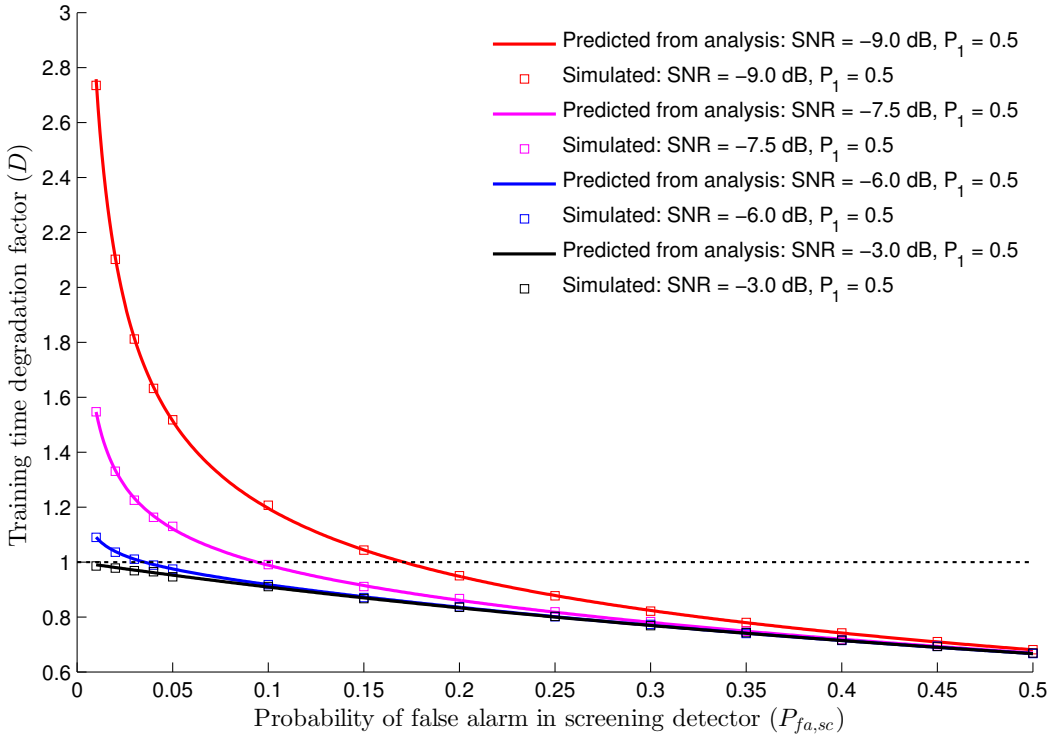

FIGURE 3.2: The effect on average training time as defined by the training time degradation factor, as $P_{fa,sc}$ varies, for low SNR; $P_0 = P_1 = 0.5$.

3.3 Simulation Results: Detection Performance

As described in section 3.1.3, it seems intuitive that, as $P_{fa,sc}$ increases, the feature-based detector's performance will deteriorate, as more false inclusions are made during its training and feature identification process. To illustrate this empirically, a MATLAB simulation was established, whereby a simple learning scheme was used to identify the cyclostationarity features present in the licensed network user's signal (namely, its

Parameter	**Description**
Observation length, N samples	256
Training set size, N_{ts}	32
Screening detector	Energy detector
Licensed network user's signal	OFDM, 1024 carriers per symbol, unmodulated pilot carrier with amplitude 7 repeated every 128 carriers, guard interval $\frac{1}{8}$ symbol with cyclic prefix

TABLE 3.1: Training time simulation parameters

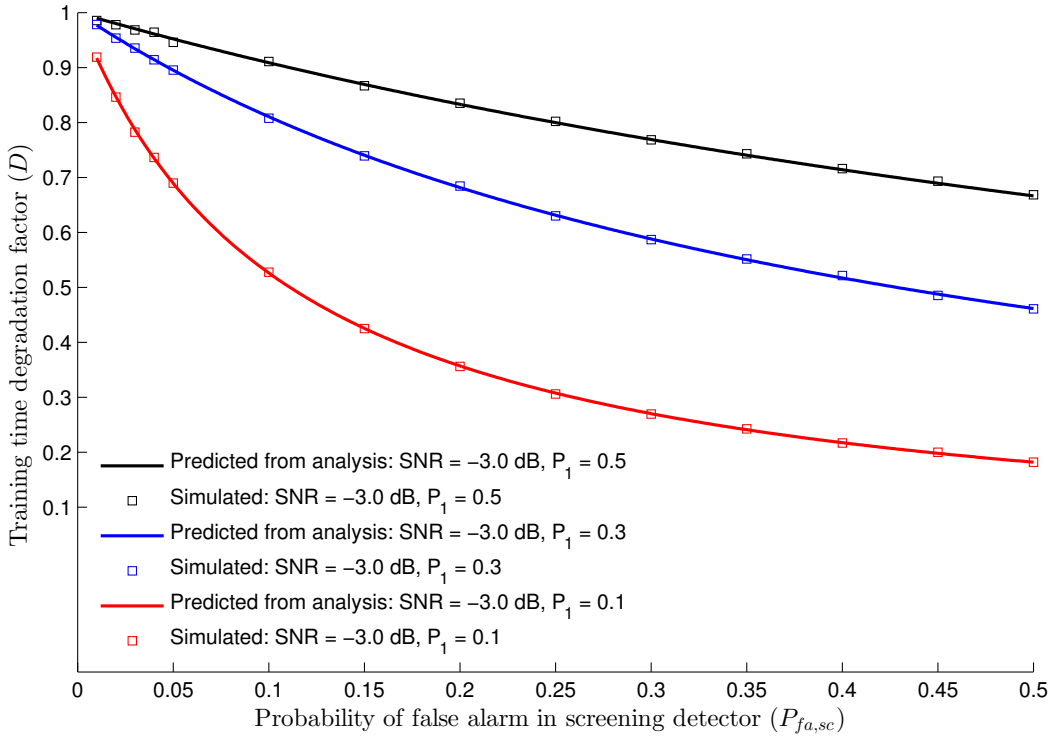

FIGURE 3.3: The effect on average training time as defined by the training time degradation factor, as P_1 varies, for SNR = -3 dB.

SCD).

The learning scheme can be described as follows:

1. For each of the N_{ts} signal observations labelled with $L = 1$ and admitted into the training set, an SCD estimate was carried out using FAM (as described in section 2.3.2). The *magnitude* of the SCD estimate was then taken.

2. All N_{ts} SCD magnitude estimates were then summed together.

3. From the summed SCD magnitude estimates, M maximal values were identified, and their locations on the $f - \alpha$ plane became the *cyclostationarity features* of the signal of interest. These locations make up a set of co-ordinates, ϕ_1.

In relation to step 3, a genetic algorithm was used to identify the peak locations in the cumulative SCD magnitude estimate, similar to the approach described in section 2.6.1. However, any peak identification algorithm could conceptually be used, including an

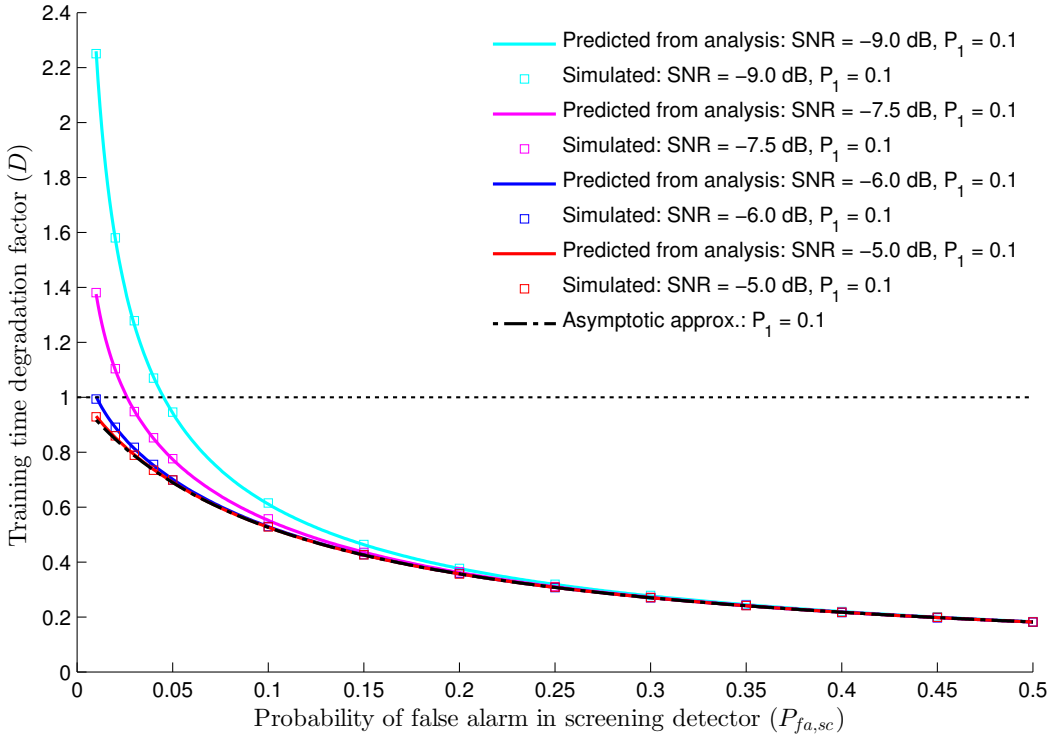

FIGURE 3.4: The effect on average training time as defined by the training time degradation factor, as SNR varies, and comparison to asymptotic expression (3.10); $P_1 = 0.1$.

exhaustive sort operation on each point in the cumulative SCD magnitude estimate – a point that was mentioned in section 2.7. FIGURE 3.5 illustrates these steps.

FIGURE 3.6 shows an example of identified features obtained via simulation, as a result of the training process, for two cases – when $P_{fa,sc} = 0.05, 0.5$. The licensed network user of interest was assumed to be transmitting an OFDM signal with eight evenly-spaced pilot subcarriers per symbol, as used in section 2.6.2. The M feature points in ϕ_1 are shown as dots – dark red dots illustrate the top quartile of points in terms of the cumulative SCD magnitude, light red dots mark the next quartile, and the remaining points are indicated with yellow-orange dots. The $P_{fa,sc} = 0.05$ case is closer to the perfect training scenario, and hence the identified signal features, due to the OFDM pilots, more closely match the pattern that would be expected from such a signal. (Refer to FIGURE 2.16 for an illustration of the expected SCD pattern.)

For the purposes of this exercise, it was assumed that all points where $\alpha = 0$ were

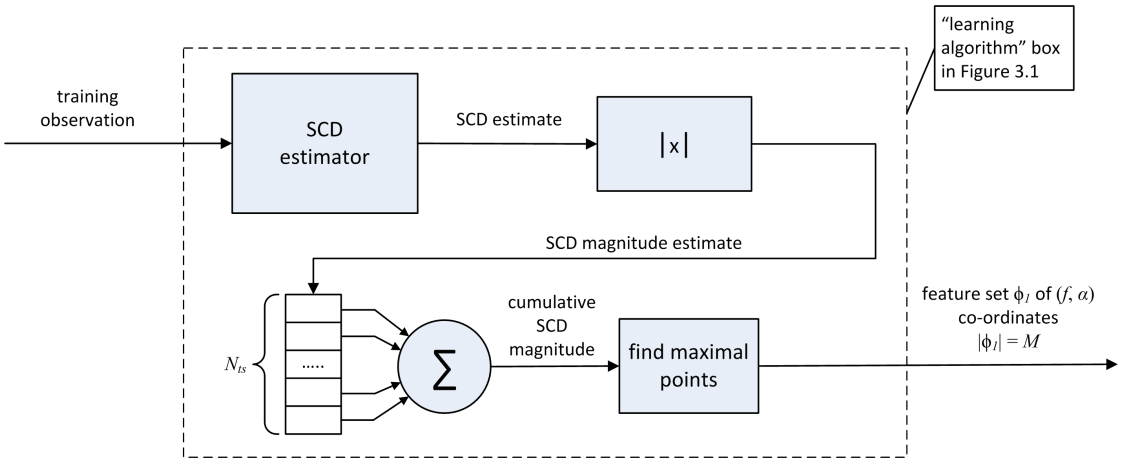

FIGURE 3.5: Simple training algorithm to identify features in the spectral correlation density of a licensed network user's transmission.

Parameter	**Description**
SNR	-3 dB
P_0, P_1	0.9, 0.1
Maximal points selected after training, M	256
Training and signal parameters are the same as in TABLE 3.1.	

TABLE 3.2: Post-training feature identification and detection simulation parameters.

included into a set of points, ϕ_0:

$$\phi_0 = \{(f, \alpha) \,|\alpha = 0\}\,. \tag{3.11}$$

The set of *all* points, used by the detector in its operational phase after training, is then given by Ω, defined as per (2.51). During operation, when a detection decision needs to be made for a signal observation, (2.52) is applied.

With these guidelines in place, a cyclostationarity detector's detection-error performance was evaluated, for different values of $P_{fa,sc}$ used during training, resulting in several performance curves as illustrated in FIGURE 3.7. Some parameters for the training and subsequent detection simulation are listed in TABLE 3.2. A low value of P_1 was used, representing a relatively underutilized channel (one where cognitive radio technology is considered to be most useful).

From FIGURE 3.7, there is perceptible detection-error performance deterioration

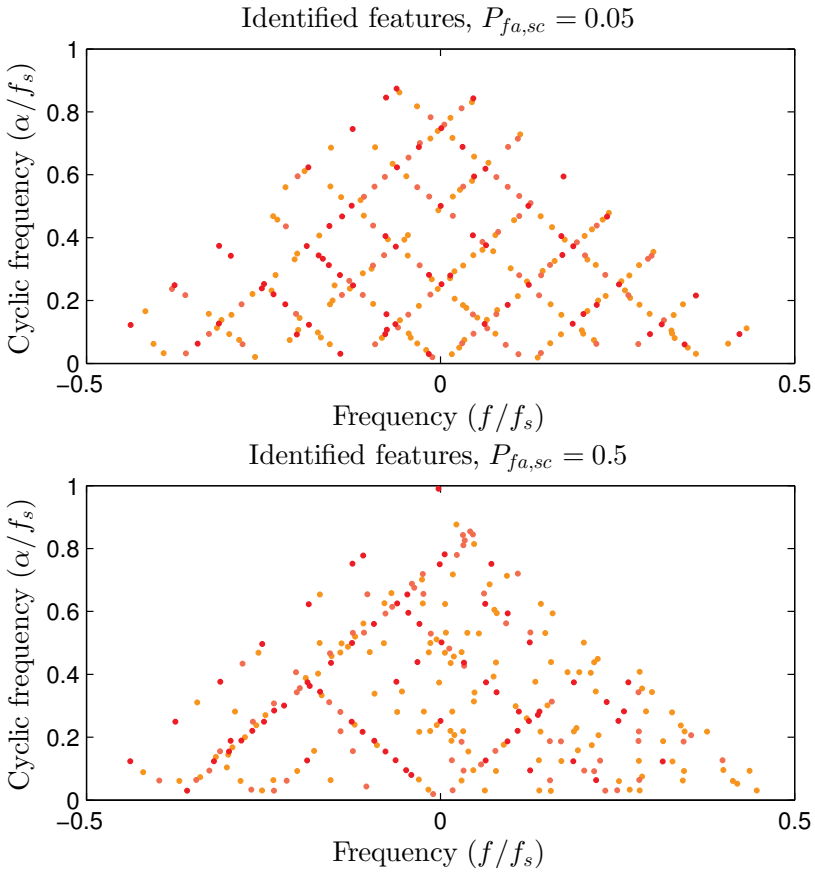

Figure 3.6: Cyclic spectral features for an OFDM signal with pilot carriers, at two values of $P_{fa,sc}$. Such a signal would normally have peaks located at the points of intersection of lines in a diagonal grid pattern. The features identified by the adaptive system are less accurate when $P_{fa,sc} = 0.5$ compared to 0.05, as indicated by the presence of fewer feature points lying on the diagonal grid and a general 'smearing' of the feature locations on the $f - \alpha$ plane. This is due to a larger amount of false inclusions during training.

due to imperfect training, as $P_{fa,sc}$ increases. The sensitivity of the detection performance curves to $P_{fa,sc}$ will depend on the relative 'strength' of the features that cyclostationarity analysis can extract from the transmission of the licensed network user of interest as compared to noise.

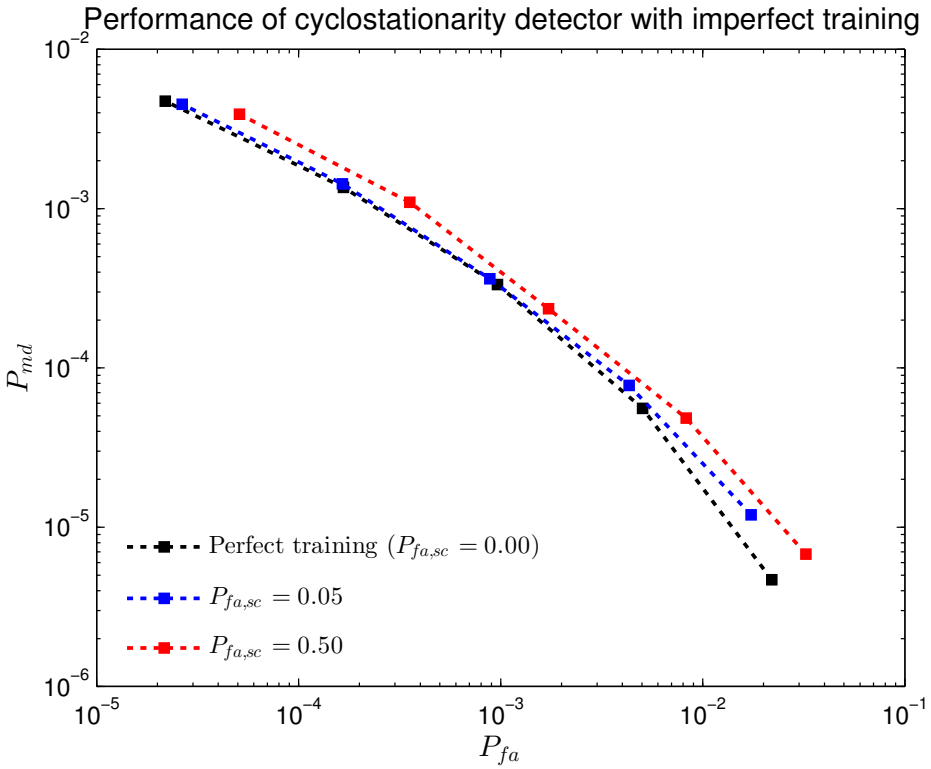

FIGURE 3.7: Cyclostationarity detector performance for different values of $P_{fa,sc}$.

3.4 Training Time and Detection Performance Trade-off Considerations

Sections 3.2.3 and 3.3 together confirm the trade-off between training time and detection-error performance as $P_{fa,sc}$ varies – decreasing it leads to longer training times but better detection performance.

Conceptually, this trade-off can be used to formulate an optimization problem, whose solution can guide us towards determining an appropriate value of $P_{fa,sc}$ (and hence the threshold, $\lambda_{T,sc}$) of the screening detector. The function to optimize is a *cost function*. The cognitive radio will incur costs during its training and operational phases:

- *Cost of training*, C_{tr}. e.g. opportunity cost on data transmission.
- *Cost of false alarm*, C_{fa}. Not exploiting an available spectral opportunity. e.g.

opportunity cost on data transmission.

- *Cost of missed detection*, C_{md}. Results in interference being exerted on a licensed network user. e.g. (possibly financial) penalty for interfering with an incumbent service provider.

Let the average time spent on training be τ_{tr}. From (3.8), this is a function of $\lambda_{T,sc}$. If it is assumed that the cost of training is proportional to the training time, then:

$$\begin{aligned} C_{tr} &\propto \tau_{tr} \\ &= k_1 N_{it} \\ &= k_1 D \left(\frac{N_{ts}}{P_1} \right), \end{aligned} \tag{3.12}$$

where k_1 is a proportionality constant.

Once training has been completed, the cognitive radio operates for some time, τ_{op} (before it may need to be retrained to cater for, for example, a different licensed network user in a new geographical location). Over this duration, on average, the channel will be occupied by the licensed network user for time $P_1\tau_{op}$ (and vacant for time $P_0\tau_{op}$). Assuming the cost of false alarms and missed detections are proportional to the operational time:

$$C_{fa} = k_2 P_{fa} P_0 \tau_{op}, \tag{3.13}$$

$$C_{md} = k_3 P_{md} P_1 \tau_{op}, \tag{3.14}$$

where k_2 and k_3 are proportionality constants.

The total cost function then becomes:

$$\begin{aligned} C &= C_{tr} + C_{fa} + C_{md} \\ &= k_1 D \left(\frac{N_{ts}}{P_1} \right) + k_2 P_{fa} P_0 \tau_{op} + k_3 P_{md} P_1 \tau_{op}. \end{aligned} \tag{3.15}$$

From (3.9), D is a function of P_1, $P_{fa,sc}$ and $P_{md,sc}$, and $P_{fa,sc}$ and $P_{md,sc}$ in turn are functions of $\lambda_{T,sc}$. Therefore, the total cost, C, can be written explicitly as a function of $\lambda_{T,sc}$:

$$C\left(\lambda_{T,sc}\right) = \left(\frac{k_1 N_{ts}}{P_1}\right) D\left(P_1, \lambda_{T,sc}\right) + \left(k_2 \tau_{op}\left(1 - P_1\right)\right) P_{fa}\left(\lambda_{T,sc}\right) + \left(k_3 \tau_{op} P_1\right) P_{md}\left(\lambda_{T,sc}\right). \tag{3.16}$$

A cognitive radio will set the threshold λ_T of its detector during operation, depending on a set of criteria which must be met. If it is set for fixed P_{fa} (i.e. constraint is placed on the probability of missing a spectral opportunity), then C_{fa} is fixed and, from an optimization perspective, can be excluded from the cost function. In this case, it is noted that P_{md} will also be a function of $\lambda_{T,sc}$ given a fixed P_{fa} (since different values of $\lambda_{T,sc}$ result in different performance curves in FIGURE 3.7). The optimization problem can now be stated as:

$$\arg\min_{\lambda_{T,sc}} \left(\left(\frac{k_1 N_{ts}}{P_1}\right) D\left(P_1, \lambda_{T,sc}\right) + \left(k_3 \tau_{op} P_1\right) P_{md}\left(\lambda_{T,sc}\right) \right). \tag{3.17}$$

Similarly, if λ_T is set for a fixed P_{md} (i.e. constraint is placed on the probability of exerting interference on a licensed network user), then P_{fa} will be a function of $\lambda_{T,sc}$ given a fixed P_{md}, resulting in:

$$\arg\min_{\lambda_{T,sc}} \left(\left(\frac{k_1 N_{ts}}{P_1}\right) D\left(P_1, \lambda_{T,sc}\right) + \left(k_2 \tau_{op}\left(1 - P_1\right)\right) P_{fa}\left(\lambda_{T,sc}\right) \right). \tag{3.18}$$

In FIGURE 3.7, the performance curves showing the relationship between P_{fa} and P_{md} were obtained via simulation – it would be useful to obtain expressions, either analytical or empirical, for $P_{fa}\left(\lambda_{T,sc}\right)$ or $P_{md}\left(\lambda_{T,sc}\right)$, if it was necessary to perform this optimization.

3.5 Concluding Remarks

In this chapter, the issue of imperfect training was highlighted in the context of a cognitive radio performing dynamic spectrum access. In the previous chapter, it was shown that a feature-based detector such as that based on cyclostationarity requires training in order to recognize and 'tune itself' to the cyclostationarity properties of a licensed network user's transmission, and this was conducted via supplying a sufficient number of training observations that were *known* to be representative of the signal which must be detected. Conducting this process out in the field, however, introduces uncertainty into this process, as the training observations must come from the real-world radio environment itself, resulting in imperfect training where not all observations are guaranteed to be truly representative of the signal of interest.

A two-stage detection model was presented to conduct this imperfect training – a screening detector was used to isolate the required training set of observations during the training phase, and then a feature-based detector that uses the training outcome to perform spectrum sensing in the operational phase. Its operation was simulated for the specific case of an energy detector as the screening detector, and a cyclostationarity detector employing SCD estimates as the operational detector.

A trade-off was also identified between the time spent in training and the resulting detection performance. In general, using a 'stricter' screening detector results in a training process that is less imperfect, and therefore the resulting trained feature detector achieves superior detection-error performance, but the time spent in the training phase is longer. Initial steps towards formulating an optimization for balancing these two objectives were also explored. The measure of 'strictness' of the screening detector used was its probability of false alarm, $P_{fa,sc}$ – the resulting cyclostationarity detection-error performance as a function of this (its probability of false alarm and missed detection) was observed empirically. The exact nature of these functions is dependent on the detection algorithm used and the features that are present in the signal of interest.

4

Search: Identify Vacant Channels with Arbitrary Boundaries

Up to this point, it has been demonstrated that the exploitation of computational resources in signal analysis and machine learning can result in more accurate spectrum sensing in cognitive radios performing dynamic spectrum access. All the approaches have operated on discrete-time signal observations, taken from the radio environment. The progenitorial signals to these observations are transmissions at RF frequencies. It is well-known from the Nyquist sampling criterion that, in order to represent any signal in discrete-time form, it must be sampled at a rate that is at least twice as high as the highest-frequency component in the signal. Analyzing a discrete-time representation of such RF signals directly would require analogue-to-digital converters (ADCs) with such high sampling rates as to be impractical. Such signals are therefore *down-converted* to

baseband frequencies, which are manageable by ADCs, for discrete-time analysis by cognitive radios, just as this step is necessary for the process of signal demodulation by the intended receiver of the signal in the licensed network.

This down-conversion process to baseband requires an understanding of both the centre frequency of the original RF signal in the radio environment and its bandwidth. Not knowing the centre frequency will result in a down-converted signal that is spectrally off-centre and adds an additional parameter of uncertainty that must be catered for in the signal detection algorithm. Not knowing the bandwidth may result in spectral information, such as energy or even *feature* information, in the original signal being lost, if the anti-aliasing filter prior to the ADC has a smaller bandwidth than the signal itself.

The existing literature on spectrum sensing frequently makes an underlying assumption that the radio environment is *channelized* in a known way – in other words, the signals of interest will adhere to a specific set of centre frequencies and bandwidths based on conventions or rules prescribed by regulatory authorities. Television broadcasts, for example, exemplify this – any television tuner knows where in the allocated spectrum to scan for the presence of broadcasts because the associated channels are specified and built in. In [57], it is assumed that the radio environment in which cognitive radios operate is subdivided into equal-spaced, equal-width frequency bands, and so cognitive radios can exploit this channelization knowledge.

There is an emphasis in the literature on cognitive radios *co-existing* with licensed network users, with the latter adhering to spectrum allocation and management schemes set by the regulatory authorities. In general, the assumption that the licensed network users' transmissions follow known channelizations seems reasonable under these circumstances. Initial cognitive radio applications will likely operate according to this, such as television whitespace devices based on IEEE 802.22 standards. However, the broader assumption that all spectral opportunities will be of equal width is overly restrictive and not useful for cognitive radio applications. Conceptually, cognitive radios should have unfettered flexibility in terms of the decisions they make on

the modulation schemes, transmission bandwidths and other communication parameters they employ. Furthermore, the next logical step in the spectrum sensing problem is that cognitive radios might want to sense the presence of *other cognitive radios* operating in the radio environment, who are free to exploit their operational flexibility and whose transmissions may not adhere to pre-existing channelizations in the way that licensed network users do. These points motivate the notion of cognitive radios sensing the presence of signals subject to *arbitrary channelizations*, where no prior assumptions are made about a signal's centre frequency and bandwidth in the radio environment.

In this chapter:

- How the traditional energy detection model can be extended, to make decisions not only on a signal's presence or absence, but also which spectral subset of the radio environment it occupies, is considered

- The existing performance measures of probability of false alarm, and probability of missed detection, are refined

- A low-complexity (in terms of computation) peak detection algorithm is presented that is the cornerstone of the extended detection model

- An empirical investigation into the detection performance of the model is presented, and some insights into what factors affect its performance are developed

- A high-complexity peak detection algorithm, based on *Bayesian decision* criteria, is presented for a very specific (but highly unrealistic) signal detection scenario, whose performance is presented as a reference

4.1 Extending Signal Detection for Arbitrary Channelizations

4.1.1 Rationale and Development of a Generalized Detection Model

The following terms are defined:

- The *sensing bandwidth* is the exploitable range of frequences of a cognitive radio for its own communication needs
- A *sensing channel* refers to a range of frequencies, making up a subset of the sensing bandwidth, from which a cognitive radio takes a signal observation (for the purpose of spectrum sensing/signal detection)

As alluded to earlier, it is a common assumption that the sensing bandwidth is divided up into equally-spaced sensing channels, and that the sensing channels align, in terms of centre frequency and bandwidth, with the transmissions of the licensed network users operating in the vicinity. In the event that there is misalignment between the two, it is possible that:

- Only part of the spectral energy of a licensed network user's transmission may fall within a sensing channel's bandwidth. This reduces the average value of the detection statistic (e.g. the observed signal energy), increasing the probability of a missed detection event. (In other words, should a cognitive radio subsequently attempt to transmit on that channel, it would cause at least partial interference with the licensed network user's transmission.)
- All of the spectral energy of a licensed network user's transmission may fall within a sensing channel, but the bandwidth of the latter is wider. Even if the cognitive radio correctly detects the presence of the licensed network user's transmission, there is still a partial spectral opportunity that has been missed, since the entire sensing channel is subsequently avoided when some of the associated spectrum is actually exploitable.

The above considerations also apply for cyclostationarity detection; spectral mismatch of the type indicated may cause the identified features on the SCD estimates to be shifted off-centre on the $f - \alpha$ plane, or be reduced in magnitude, leading to increased probability of detection errors and interference with licensed network users.

However, in order for wireless transceivers of the future to be truly cognitive and flexible, and for the technology to be able to meet its promise of more efficient exploitation of radio spectrum, they will need to be able to operate autonomously in radio environments that are channelized more arbitrarily. The ability to do so opens up the number of 'degrees of freedom' with which a cognitive radio can operate to meet its communications objectives.

Traditional energy detection models are often depicted in the time domain, but (2.14) shows that the task can also be carried out in the frequency domain. Where the channelization of licensed network users is known by the cognitive radio and therefore aligned with its sensing channels, it can repeatedly perform signal detection on all sensing channels to build up a picture of overall spectral occupancy.

The next logical step pertains to radio environments where a cognitive radio is unable to safely presume any channelization structure. It must then determine the channelization itself. To achieve this, referring to (2.7), the nature of the decision D must be extended beyond a binary 'yes/no' decision. In the event that the detector decides that H_1 is true, the decision needs to also provide an estimate of the spectral occupancy (i.e. the lower and upper spectral boundaries) of the detected signal transmission, and thus its channelization. This could be done as follows:

$$\begin{aligned} &D = [D_L \ldots D_U] \text{ (interval)} \\ &\text{If } D = \varnothing, \text{ detector decides that } H_0 \text{ is true.} \\ &\text{If } D \neq \varnothing, \text{ detector decides that } H_1 \text{ is true.} \end{aligned} \tag{4.1}$$

This is illustrated in Figure 4.1. The input to this model is a wideband signal S_W (occupying the sensing bandwidth) rather than a narrowband signal S_N (occupying

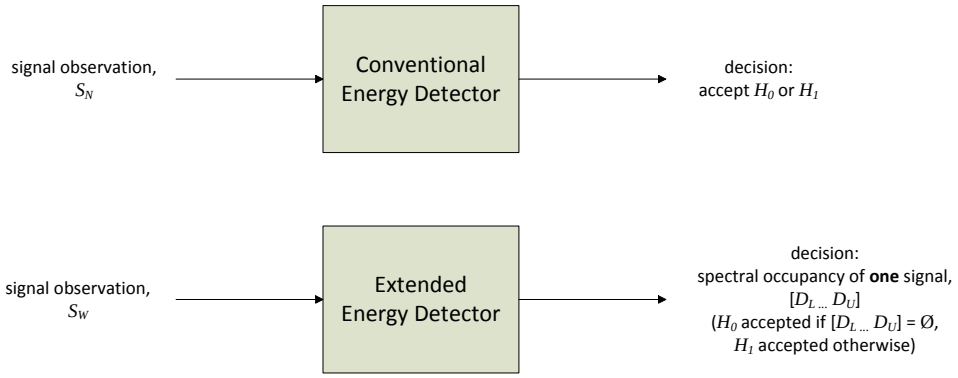

FIGURE 4.1: Comparison of a conceptual extended energy detector, which returns occupancy information, with a conventional energy detector.

the width of a presumed sensing channel). Conceptually, S_W can be obtained through a wideband receiver, or an amalgam of several spectrally-separated narrowband observations operating in parallel.

4.1.2 Model Requirements

In principle, as the generalized energy detector is required to isolate the spectral positioning of a signal within a sensing bandwidth of interest, it operates along similar principles to a frequency-domain energy detector. It can be described as follows:

1. Obtain a PSD estimate across the entire sensing bandwidth, consisting of frequency bins labelled from 1 to K. Call this PSD function $S[k]$, where k is the frequency-bin index.

2. Assume, to begin with, that there is a signal being transmitted by a licensed network user within the sensing bandwidth. Based on this assumption being true, obtain decision estimates for the lower (D_L) and upper (D_U) channel boundaries of such a signal, using the most compelling evidence supporting this assumption. D_L and D_U are frequency-bin indices, such that: $1 \leq D_L \leq D_U \leq K$.

3. Calculate the mean PSD value taken from all frequency bins between D_L and

D_U inclusive, as a test statistic, λ:

$$\lambda = \frac{1}{D_U - D_L + 1} \sum_{k=D_L}^{D_U} S[k]. \tag{4.2}$$

4. Compare λ to a threshold, λ_T. If it is less than the threshold, define: $[D_L \dots D_U] = \varnothing$.

5. Make decision D according to (4.1).

Step 2 is expanded upon in section 4.2.

For a multi-user radio environment, it is possible to use the above steps in an iterative fashion, by gradually decomposing the sensing bandwidth into spectral 'chunks', where contiguous frequency bins for which no decision has been made are iteratively fed back to the detector. Each detector iteration results in more frequency bins being accounted for by an occupancy decision, until there are none left. This is summarized in algorithm 4.1.

4.1.3 Extended Definitions of Detection Error Events

In traditional energy detection, a decision about the presence or absence of a signal applies to an entire pre-defined sensing channel, and the detection-error events of false alarm and missed detection can occur, as described in section 2.1.1.

For the extended model, these concepts need to be refined. Because a decision is made not only on the occupancy of a channel, but also on the boundaries of this occupancy as well, the boundaries are also subject to error. The implication is that a single sensing decision may actually result in subsets of the sensing bandwidth being subject to both false alarm and missed detection simultaneously.

Let the 'reality' of the actual spectral occupancy of a signal of interest be:

$$[A_L \dots A_U], \tag{4.3}$$

Algorithm 4.1 Iterative, multi-user detection across a bandwidth of interest.

Require: A set of frequency bins $[1 \dots K]$ with undetermined occupancy
  $\mathbf{S} \leftarrow \{[1 \dots K]\}$ {*Set of bin intervals to search*}
  $\mathbf{C} \leftarrow \varnothing$ {*Set of bin intervals that may be occupied*}
  while $\mathbf{S} \neq \varnothing$ **do**
    $\mathbf{S_{new}} \leftarrow \varnothing$ {*Set of bin intervals to search in next iteration*}
    for all intervals $\mathbf{k}$, where $\mathbf{k} \in \mathbf{S}$ **do**
      Find D_L, D_U within $\mathbf{k}$ {*Step 2, section 4.1.2*}
      Add interval $[D_L \dots D_U]$ to $\mathbf{C}$
      $\mathbf{r} \leftarrow [1 \dots (D_L - 1)]$
      if $\mathbf{r} \neq \varnothing$ **then**
        Add $\mathbf{r}$ to $\mathbf{S_{new}}$
      end if
      $\mathbf{r} \leftarrow [(D_U + 1) \dots |\mathbf{k}|]$
      if $\mathbf{r} \neq \varnothing$ **then**
        Add $\mathbf{r}$ to $\mathbf{S_{new}}$
      end if
    end for
    $\mathbf{S} \leftarrow \mathbf{S_{new}}$
  end while

  for all intervals $\mathbf{k}$, where $\mathbf{k} \in \mathbf{C}$ **do**
    Decide if the channel represented by $\mathbf{k}$ is occupied or not {*Steps 3-4, section 4.1.2*}
  end for

where A_L and A_U represent the *actual* lower and upper spectral boundaries, respectively, of a signal of interest. $[A_L \dots A_U] = \varnothing$ if the licensed network user is not actually transmitting during the observation time.

The following may now be defined:

- **False-alarm spectral set**: The set of frequency bins that are not actually occupied, but are deemed to be occupied by the detector. *Part of the sensing bandwidth that is available for use is not exploited.* When $[A_L \dots A_U] = \varnothing$, this can affect frequency bins $1 \leq k < A_L$ and $A_U < k \leq K$. Described more formally as:

$$\mathbf{S_{fa}} = [D_L \dots D_U] \setminus [A_L \dots A_U], \tag{4.4}$$

  where $\mathbf{A}\backslash\mathbf{B}$ is the *relative complement* of set $\mathbf{B}$ in $\mathbf{A}$.

- **Missed-detection spectral set**: The set of frequency bins that are actually

occupied, but are deemed to be vacant by the detector. *In exploiting spectrum accordingly, some interference may result upon another radio user.* When $[A_L \dots A_U] = \varnothing$, this affects frequency bins where $A_L \leq k \leq A_U$. This may be described more formally as:

$$\mathbf{S_{md}} = [A_L \dots A_U] \setminus [D_L \dots D_U]\,. \tag{4.5}$$

A decision may be non-empty in both $\mathbf{S_{fa}}$ and $\mathbf{S_{md}}$ simultaneously.

A *perfectly correct* decision results in:

$$\mathbf{S_{fa}} = \mathbf{S_{md}} = \varnothing. \tag{4.6}$$

On the other hand, a *completely incorrect* decision occurs when the following is logically true:

$$\begin{aligned}&(\mathbf{S_{fa}} = [D_L \dots D_U]) \wedge (\mathbf{S_{md}} = [A_L \dots A_U]) \wedge \\ &(([D_L \dots D_U] \neq \varnothing) \vee ([A_L \dots A_U] \neq \varnothing))\,.\end{aligned} \tag{4.7}$$

Completely incorrect decisions can be one of the following scenarios:

- The signal of interest is not present, but D is non-empty

- The signal of interest is present, but D is empty

- The signal of interest is present, and D is non-empty, but the actual occupancy spectral interval $[A_L \dots A_U]$ and the decision spectral interval $[D_L \dots D_U]$ do not overlap.

4.2 Spectral Peak Detection Using Maximum Slope Magnitudes

In section 4.1.2, Step 2 requires a method of estimating the lower and upper channel boundaries of a signal which may be present. In principle, some form of peak detection algorithm is required, to isolate an area of higher spectral energy than its surroundings.

A low-complexity approach is described here, that identifies the locations of significant upward and downward transitions (slopes) in the PSD taken across the sensing bandwidth. In principle, this approach is akin to calculating the first derivative of the PSD function, and finding where the maximum (rising edge) and minimum (falling edge) slopes occur. It can be considered similar to finding the inflexion points of a continuously differentiable function by considering points where the second derivative is zero.

The algorithm can be summarized by the following steps.

1. Apply a smoothing (averaging) window to $S[k]$ by performing a convolution with a rectangular window of unity magnitude and width of W_0 frequency bins, resulting in a new function $S_S[k]$. ($W_0 = 1$ is equivalent to no smoothing.)

2. Calculate a slope function $S'[k]$, by calculating the difference between adjacent points of $S_S[k]$:
$$S'[k] = S_S[k] - S_S[k-1]. \tag{4.8}$$

3. Apply a smoothing window to $S'[k]$ by performing a convolution with a rectangular window of unity magnitude, where the width of the window is W_1 frequency bins, resulting in a new function $S'_S[k]$. ($W_1 = 1$ is equivalent to no smoothing.)

4. Assuming that H_1 is true, estimate the centre frequency, D_C, of the signal of interest, by locating the maximal value of $S_S[k]$, and the corresponding frequency bin:
$$D_C = \underset{k \in [1 \ldots K]}{\arg\max}\, S_S[k]. \tag{4.9}$$

5. Still assuming that H_1 is true, estimate lower boundary D_L by locating the maximal value of $S'_S[k]$, and the corresponding frequency bin, constrained by $k \leq D_C$.

$$D_L = \underset{k \in [1 \ldots D_C]}{\arg\max} S'_S[k]. \tag{4.10}$$

6. Still assuming that H_1 is true, estimate upper boundary D_U by locating the minimal value of $S'_S[k]$, and the corresponding frequency bin, constrained by $k \geq D_C$.

$$D_U = \underset{k \in [D_C \ldots K]}{\arg\min} S'_S[k]. \tag{4.11}$$

It is expected that this approach would be particularly suited for signals whose power spectral density is rectangular in shape (e.g. OFDM transmissions).

4.2.1 Simulation Results

MATLAB simulations have been used to assess the described peak detection technique, *where a signal of interest is always present in a fixed spectral location.* In a Monte Carlo simulation, estimates of D_L and D_U become random variables whose discrete probability mass functions (pmf) can be observed. *It is assumed in all these simulations that there is only one signal of interest present within the sensing bandwidth.*

The discrete pmfs can be expressed as follows:

$$f_L[k] = P(D_L - A_L = k), \tag{4.12}$$

$$f_U[k] = P(D_U - A_U = k), \tag{4.13}$$

where $P(X)$ denotes the probability of an event X occurring, and $f_L[k]$ and $f_U[k]$ define the pmfs of D_L and D_U respectively. In this context, k is the difference between the spectral boundary decision and its actual underlying value. Note that the random variables corresponding to D_L and D_U are assumed to be independent when visualizing these pmfs. (In reality this is not quite true, as the algorithm ensures that, for example, $D_U > D_L$, implying some dependence.)

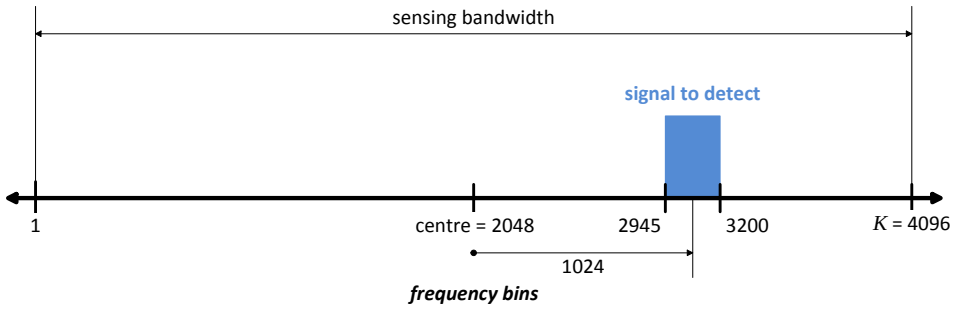

FIGURE 4.2: Spectral occupancy of the signal to be detected, within the sensing band, with a spectral offset of +1024 frequency bins.

TABLE 4.1 summarizes the simulation parameters used, the characteristics of the licensed network user's signal transmission, and the detection results obtained. Within this table, the *spread* value refers to the number of frequency bins on either side of the actual boundary value, between which 68.2% of estimates fall. In this respect, it is similar to the definition of the standard deviation as applied to Gaussian (normal) distributions, where 68.2% of values fall within one standard deviation of the mean.

FIGURE 4.2 illustrates the concept of *spectral offset* of the signal within the sensing bandwidth, in the *baseline case* (first column of TABLE 4.1).

Some observations from these results are now elaborated upon.

4.2.2 Boundary Estimate Spread - Amount

From TABLE 4.1, it is seen that a deterioration in SNR (*SNR = -3 dB*), and the presence of channel fading (*Channel fading*), all increase detection spread (variability) in the boundary estimates, compared to the *Baseline* case, as would be expected. The magnitude of the estimator bias is also increased.

On the other hand, increasing the signal observation length (*Observ. length 8192*) reduces the spread, as would be expected. Estimator bias is also reduced.

Table 4.1: Detection Simulation using Maximum Slope Magnitudes, When Licensed Network User Signal is Present

		Simulation Cases								
		Baseline case	**SNR = -3 dB**	**Spectral avg. 129 bins**	**Channel fading**	**-1024 offset from centre**	**No offset**	**No offset, pilot on upp. bound.**	**No offset, no pilots**	**Observ. length 8192**
Parameters	**No. detection events**	1000000								
	Noise	Additive White Gaussian Noise (AWGN)								
	Licensed network user signal	OFDM with symbol length 1024; guard interval 1/8 of a symbol; data sub-carriers QPSK modulated; pilot sub-carriers with amplitude 8/3								
	Pilot sub-carrier positions	1 129 257 385 ... 897	1 129 257 385 ... 897	1 129 257 385 ... 897	1 129 257 385 ... 897	1 129 257 385 ... 897	1 129 257 385 ... 897	128 256 384 512 ... 1024	None	1 129 257 385 ... 897
	No. freq. bins (K)	4096	4096	4096	4096	4096	4096	4096	4096	8192
	Freq. occupancy $A_L \dots A_U$	2945 ... 3200	2945 ... 3200	2945 ... 3200	2945 ... 3200	897 ... 1152	1921 ... 2176	1921 ... 2176	1921 ... 2176	2945 ... 3200[1]
	SNR (dB) (in occupancy bins)	0	-3	0	0	0	0	0	0	0
	Smoothing window $W_0 = W_1$	65	65	129	65	65	65	65	65	65[2]
	Channel fading	None	None	None	Yes[3]	None	None	None	None	None
	Missed detection rate P_{md}[4]	0.01								
Results	**Probability of correct D_L ($f_L[0]$)**	0.343	0.098	0.345	0.185	0.344	0.314	0.170	0.155	0.540
	Probability of correct D_U ($f_U[0]$)	0.166	0.059	0.167	0.116	0.166	0.188	0.312	0.172	0.201
	D_L bias $(\overline{D}_L - A_L)$[5]	-1.9	-405.8	1.1	8.6	1.5	0.3	1.4	1.0	0.1[6]
	D_U bias $(\overline{D}_U - A_U)$[5]	-2.9	43.9	-2.5	-12.9	0.7	-1.3	-0.3	-0.8	-1.3[6]
	Spread s_{D_L}	6	65	6	11	6	6	6	6	2[6]
	Spread s_{D_U}	6	27	6	12	6	6	6	6	3[6]
	Perfectly correct (%)[7]	5.8	0.6	5.7	2.1	5.8	5.7	5.3	2.6	10.9
	Completely incorrect (%)[7]	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	False-alarm spectral set (%)[7]	51.0	76.2	49.6	44.2	51.0	51.2	51.7	55.5	45.2
	Mean false alarm spect. set size[8]	8	598	9	17	8	8	8	8	4[6]
	Miss.-detect. spectral set (%)[7]	69.9	64.4	68.8	83.6	69.9	70.0	70.7	75.3	63.0
	Mean miss. det. spect. set size[8]	14	35	13	36	14	14	14	14	6[6]

[1] As there are 8192 bins, actual occupancy is 5890 ... 6400. Bin width value here is normalized to the *Baseline* case with 4096 bins, so value is halved.
[2] Actual windows had width of 129 frequency bins. Bin width value here is normalized to the *Baseline* case, so value is halved.
[3] Fading model: JTC Outdoor Residential Low Antenna, Type C; Doppler shift $10^{-7}\times$ propagation frequency.
[4] The detection threshold λ_T was set such that the proportion of detection events where the test statistic λ was less than this (resulting in an incorrect H_0 decision), equalled P_{md}.
[5] Here, *bias* is the difference between the *sample mean* of the estimated value, calculated over all detection events, and the actual value.
[6] As there are twice as many bins as the other cases, bin width value here is normalized to the *Baseline* case, so actual value is halved to obtain this.
[7] These percentages are calculated from those detection events where the test statistic λ was above the detection threshold λ_T (resulting in H_1 decisions). Also refer to section 4.1.3.
[8] *Size* refers to the number of frequency bins in $\mathbf{S_{fa}}$ or $\mathbf{S_{md}}$ — refer to section 4.1.3.

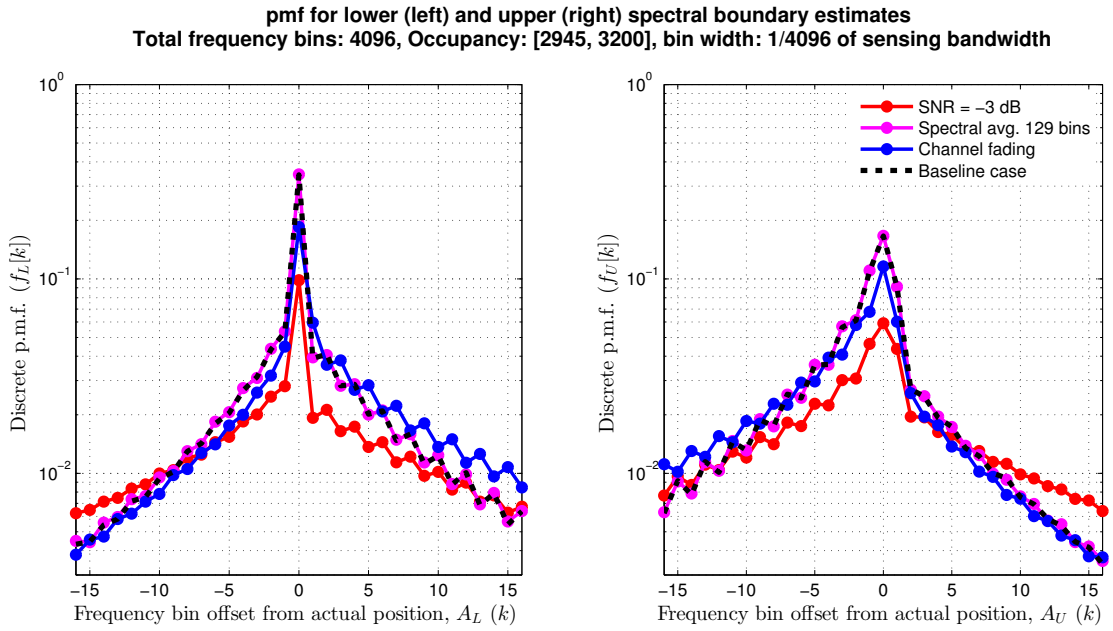

FIGURE 4.3: pmfs for the lower and upper spectral-boundary estimates under various conditions of channel impairment and smoothing scenarios.

4.2.3 Boundary Estimate Spread - Asymmetry

From TABLE 4.1, in the *SNR = -3 dB* case, the spread in the estimate of D_L is significantly larger than that of D_U. The corresponding estimator biases are also significantly different.

An explanation for this is that there are more 'candidate' frequency bins below D_L, than above D_U, in the sensing bandwidth, due to the spectral offset – hence, there are more 'opportunities' to make an erroneous D_L determination, with greater disparity from the true value, compared to D_U.

4.2.4 Boundary Estimate pmf Shapes

By considering FIGURE 4.3, it is apparent that a deterioration in SNR (*SNR -3 dB*), and the presence of channel fading (*Channel fading*), all adversely affect the detector's ability to identify the spectral boundaries reliably (compared to the *Baseline* case), since the corresponding pmfs exhibit a lower peak value (at $k = 0$) and a gentler roll-off on either side.

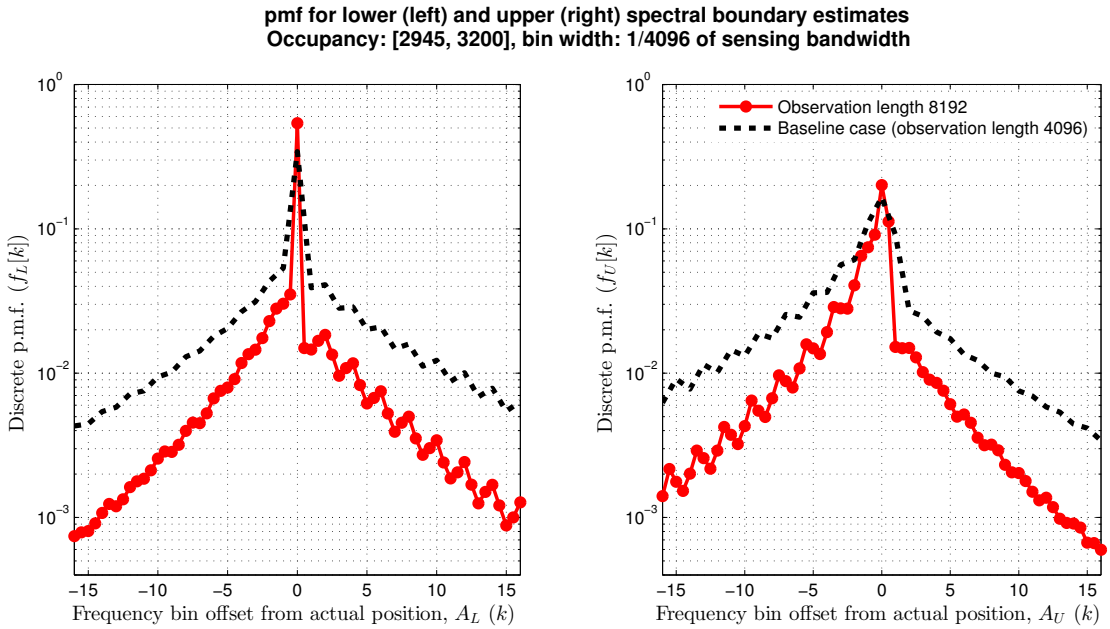

Figure 4.4: pmfs for the lower and upper spectral boundary estimates with different observation lengths. Bin offsets for the *Observation length 8192* case have been normalized to the *Baseline* case.

Conversely, From Figure 4.4, we see an improvement with a longer observation length (*Observ. length 8192*), as indicated by the higher peak values (at $k = 0$) corresponding to correct decisions on the spectral boundaries.

4.2.5 Boundary Estimate pmf Asymmetry

From Figure 4.3, it is apparent that for the *Baseline* case, the pmfs $f_L[k]$ and $f_U[k]$, as well as the estimator bias values, are asymmetric. In particular, the pmf for $f_L[k]$ has a higher peak value (at $k = 0$) and steeper roll-off; in other words, the lower boundary is detected more reliably than the upper boundary. This is due to the presence of a pilot subcarrier on the lower boundary (and its absence on the upper boundary), which carries more power than the data-bearing subcarriers. *As the peak detector uses a slope-based algorithm, the pilot actually enhances the slope at the lower boundary and assists the detector.*

Figure 4.5 illustrates what happens when the positions of the pilot subcarriers are determined by *counting downwards from the highest-frequency subcarrier*, rather

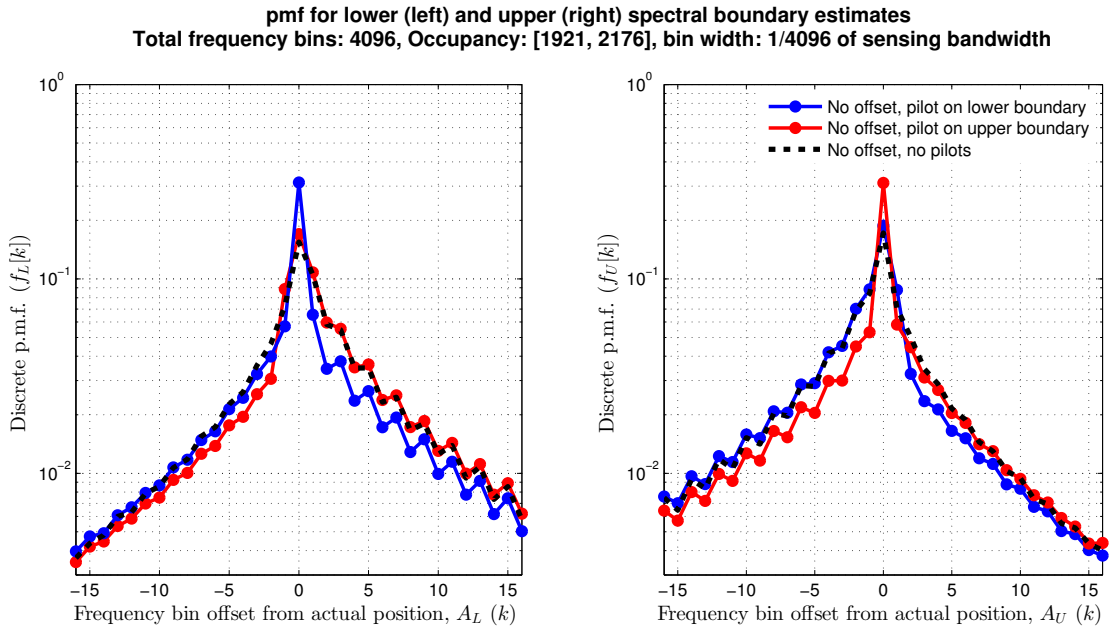

FIGURE 4.5: pmfs for the lower and upper spectral boundary estimates can be affected by the location of pilot subcarriers - placing a pilot on the spectral boundary assists in its estimation.

than counting upwards from the lowest-frequency subcarrier (*No offset, pilot on upp. bound.*) — the asymmetry is reversed. Furthermore, when there are no pilots, neither boundary is favoured, which removes the asymmetry and equalizes the bias.

4.2.6 Other Occupancy Detection Results

The simulation results in TABLE 4.1 also show the number of detection events that fell into each of the categories described in section 4.1.3.

It can be seen that:

- For poorer SNR (*SNR = -3 dB*), the number of perfectly correct detection events is reduced, and there is a deterioration in the number of both completely incorrect detections and missed detections.

- In the presence of additional channel fading perturbation (*Channel fading*), the number of perfectly correct detections degenerates, and there is deterioration in events subject to false alarms.

- When the pilot subcarriers are removed from the licensed network user's signal (*No offset, no pilots*), there is a conspicuous reduction in the number of perfectly correct detections. (Remember from section 4.2.5 that the presence of pilots actually assists the detector in identifying the boundaries.)

- Increasing the observation time (*Observ. length 8192*) improves the number of perfectly correct detections, and reduces the number of frequency bins subject to false alarm and missed detection.

4.3 Spectral Peak Detection Using Bayesian Decision Theory

To round out this chapter, a peak-detection approach is described that makes use of principles of *Bayesian decision theory*.

In principle, this approach uses empirical evidence from the PSD estimate of a wide-band observation taken across the sensing bandwidth, as well as some assumed a-priori probabilities, to compute which scenario would have resulted in the PSD observation with *maximum likelihood*.

In the following section, the derivation of the decision statistic is presented, as well as its underlying assumptions.

4.3.1 Deriving the Bayesian Peak Detection Statistic

Suppose it is true that there is exactly one licensed network user transmitting across the sensing bandwidth at any given time. This supposition is imposed to make the problem tractable and reduce the computation involved in an implementation. (In principle, any number of users could be considered, at the cost of an explosion in the number of possible detection hypotheses, and hence the amount of computation.) This also matches the constraints imposed on simulations whose results were presented previously in this chapter.

Unlike the traditional signal detection where a decision is made between two hypotheses as described in (2.1), for a sensing bandwidth with K frequency bins, numbered from 1 to K, the detection here is subject to multiple hypotheses:

$$
\begin{aligned}
&H_0 \quad : \text{No signal was transmitted during the observation time} \\
&H_1 \quad : \text{A signal was transmitted, occupying 1 bin, starting from bin 1} \\
&H_2 \quad : \text{A signal was transmitted, occupying 1 bin, starting from bin 2} \\
&H_3 \quad : \text{A signal was transmitted, occupying 1 bin, starting from bin 3} \\
&\vdots \\
&H_K \quad : \text{A signal was transmitted, occupying 1 bin, starting from bin } K \\
&H_{K+1} : \text{A signal was transmitted, occupying 2 bins, starting from bin 1} \\
&H_{K+2} : \text{A signal was transmitted, occupying 2 bins, starting from bin 2} \\
&\vdots \\
&H_{2K-1}: \text{A signal was transmitted, occupying 2 bins, starting from bin } K-1 \\
&H_{2K} \quad : \text{A signal was transmitted, occupying 3 bins, starting from bin 1} \\
&H_{2K+1}: \text{A signal was transmitted, occupying 3 bins, starting from bin 2} \\
&\vdots \\
&H_{3K-3}: \text{A signal was transmitted, occupying 3 bins, starting from bin } K-2 \\
&\vdots \\
&H_{Y-2} : \text{A signal was transmitted, occupying } K-1 \text{ bins, starting from bin 1} \\
&H_{Y-1} : \text{A signal was transmitted, occupying } K-1 \text{ bins, starting from bin 2} \\
&H_Y \quad : \text{A signal was transmitted, occupying } K \text{ bins, starting from bin 1.}
\end{aligned}
\tag{4.14}
$$

where Y is the total number of scenarios where the licensed network user is transmitting. This value can be computed by observing from (4.14) that there are K 1-bin scenarios, $(K-1)$ 2-bin scenarios, $(K-2)$ 3-bin scenarios, etc., forming an arithmetic

series:

$$\begin{aligned} Y &= K + (K-1) + (K-2) + \ldots + 2 + 1 \\ &= \frac{K}{2}\left(2K - (K-1)\right) \\ &= \frac{K\left(K+1\right)}{2}. \end{aligned} \tag{4.15}$$

(Note that, if the problem was expanded to consider the possibility of more than one licensed network user transmitting, the number of hypotheses, Y, would be much larger than this.)

Now, suppose that, for the purposes of the peak detector, it is presumed that the licensed network user *is* transmitting somewhere within the sensing bandwidth. However, the channelization of the user's transmission in terms of spectral width and position within the sensing bandwidth is unknown. In the total absence of prior information about the channelization that will be employed by the user, a conservative assumption is made that every possible channelization scenario, where the licensed network user is transmitting, occurs with equal probability, i.e.:

$$P\left(H_1\right) = P\left(H_2\right) = P\left(H_3\right) = \ldots = P\left(H_Y\right). \tag{4.16}$$

Let the observed PSD across the sensing bandwidth be represented by a vector of values (see section 4.1.2):

$$\mathbf{s} = \left[S[1]\ S[2]\ S[3] \ldots S[K-1]\ S[K]\right]. \tag{4.17}$$

For all possible scenarios in (4.14), define the *likelihood* of scenario H_y being true, given an observed PSD $\mathbf{s}$, as:

$$P\left(H_y|\mathbf{s}\right) = \frac{P\left(\mathbf{s}|H_y\right)P\left(H_y\right)}{P\left(\mathbf{s}\right)}, \tag{4.18}$$

where the right-hand side follows from Bayes' Rule. The approach is to find the scenario

H_y which results in a *maximum likelihood* value. However, $P(\mathbf{s})$ is constant, as is $P(H_y)$ from (4.16), as it is independent of y. Therefore, with these constraints:

$$P(H_y|\mathbf{s}) \propto P(\mathbf{s}|H_y). \tag{4.19}$$

The implication here is that the maximum-likelihood criterion is satisfied by finding H_y which maximizes $P(\mathbf{s}|H_y)$. To compute this, each element of $\mathbf{s}$ is treated as a random variable that has one of two underlying *known* pdfs, depending on whether or not the licensed network user's transmission occupies the corresponding frequency bin or not in a given scenario H_y. Let these two pdfs be $f_{1,k}(S[k])$ and $f_{0,k}(S[k])$, respectively, defined for a frequency bin k.

Now, assume that every scenario H_y has a corresponding *occupancy vector*, where the kth element has a binary value, depending on whether or not the kth frequency bin in the sensing bandwidth is occupied under that scenario:

$$\mathbf{o}_y = [O_y[1]\, O_y[2]\, O_y[3] \ldots O_y[K-1]\, O_y[K]], \tag{4.20}$$

where:

$$O_y[k] = \begin{cases} 0 & \text{if frequency bin } k \text{ is not occupied under scenario } H_y \\ 1 & \text{otherwise.} \end{cases} \tag{4.21}$$

Therefore:

$$P(\mathbf{s}|H_y) = \prod_{k=1}^{K} P(S[k]|O_y[k]) \propto \prod_{k=1}^{K} f_{O_y[k],k}(S[k]). \tag{4.22}$$

Note that, because the pdfs $f_{0,k}(S[k])$ and $f_{1,k}(S[k])$ are continuous, the 'probability of $S[k]$ having a specific value' is not applicable, but is considered proportional to the value of the pdf at $S[k]$, and can be used without invalidating the final result.

With the goal being to maximize $P(\mathbf{s}|H_y)$, the decision made by the peak detector is now:

$$D = H_y, \tag{4.23}$$

such that:

$$y = \underset{y \in [1 \ldots Y]}{\arg\max} \left(\prod_{k=1}^{K} f_{O_y[k],k} \left(S[k] \right) \right), \tag{4.24}$$

and the corresponding occupancy decision is given by $\mathbf{o}_y$. Then the decisions relating to the spectral boundaries of the licensed network user's transmission are:

$$\begin{aligned} D_L &= \underset{k \in [1 \ldots K]}{\arg\min} \left(O_y \left[k \right] = 1 \right), \\ D_U &= \underset{k \in [1 \ldots K]}{\arg\max} \left(O_y \left[k \right] = 1 \right), \end{aligned} \tag{4.25}$$

and detection can proceed as described in section 4.1.2.

4.3.2 Simulation Results

Some simulations similar to those described in section 4.2.1 were performed for a detector described above.

However, in order to do this, a simulation scenario is needed where the underlying pdfs, $f_{0,k} \left(S \left[k \right] \right)$ and $f_{1,k} \left(S \left[k \right] \right)$, associated with the kth element of the PSD estimate $\mathbf{s}$, are well-defined. In order to achieve this, and without loss of generality of the approach, the following simplifying assumptions were made:

- The licensed network user's signal of interest has a rectangular (constant) DFT spectrum with in-phase and quadrature components having equal magnitude.
- The magnitude of the rectangular DFT spectrum of the licensed network user's transmission does not change for different occupancy scenarios.
- There is no frequency-selective attenuation or fading on the licensed network user's transmission as seen by the cognitive radio. Any flat attenuation between the licensed network user and the cognitive radio is exactly known (and can hence be compensated for by a simple scale factor), and the only other channel perturbation is AWGN.

- The PSD is estimated by the cognitive radio using a simple periodogram by computing the magnitude-squared at each frequency bin of a K-length FFT taken across the entire sensing bandwidth

In order to realize a rectangular DFT spectrum in its transmission, the licensed network user was assumed to employ OFDM, making use of a 256-point IFFT (the same length as the actual occupancy, in terms of number of frequency bins in the sensing bandwidth seen by cognitive radio) – each sub-carrier is data-bearing and modulated with QPSK; there are no pilot sub-carriers, nor is there any guard interval or cyclic prefix used. It is further assumed that the frequency bins in the sensing bandwidth align exactly with the OFDM sub-carriers of the licensed network user's transmission.

Under such constrained conditions, in obtaining the PDF estimate, the cognitive radio effectively 'undoes' the IFFT that the licensed network user applies at the transmitter as part of its OFDM scheme[1]. This means that the cognitive radio also sees a rectangular-shaped pulse in its PSD estimate, spanning the frequency bins occupied by the licensed network user, albeit perturbed by AWGN.

Let the discrete-time wideband signal observation used to estimate the PSD across the sensing bandwidth by the cognitive radio be $s_w[n]$. Also let σ_w^2 represent the complex AWGN power present across the entire sensing bandwidth. If the estimate of the PSD is obtained by the following squared-magnitude of the DFT:

$$S\left[k\right] = \frac{2}{K\sigma_w^2}\left|\sum_{n=0}^{K-1} s_w\left[n\right] e^{-\frac{2\pi kn}{K}}\right|^2, \tag{4.26}$$

then it can be shown (see section A.6) that each frequency bin is subject to one of the following pdfs:

[1] A known scale factor reflects the difference in length between the IFFT of the licensed network user's OFDM scheme and the FFT used in the PSD estimate.

$$\begin{aligned} &f_0\left(S\left[k\right]\right) \text{ is a central } \chi^2\text{-distribution with 2 DOF} \\ &f_1\left(S\left[k\right]\right) \text{ is a non-central } \chi^2\text{-distribution with 2 DOF,} \\ &\qquad \text{with non-centrality parameter } \delta = 4\left(\frac{|m|}{\sqrt{K}\sigma_w}\right)^2, \end{aligned} \tag{4.27}$$

where $|m|$ is the magnitude of the in-phase and quadrature part of the rectangular pulse corresponding to the occupancy of the licensed network user, seen after taking the K-point FFT of $s_w[n]$. Because QPSK was used by the licensed network user to modulate each OFDM sub-carrier, the magnitude is constant across sub-carriers as well as between the in-phase and quadrature components. Note that the pdfs do not vary with k, so it has been removed from the subscripts of the pdfs above.

As an aside, the PSD coefficients $S[k]$ in (4.26) represent the scaled-squared magnitude of a sequence of DFT 'vectors' with two (real and imaginary) components. The real and imaginary components are iid, and both have either zero mean (pdf $f_0(x)$ applies) or non-zero mean (pdf $f_1(x)$ applies). Then $\sqrt{S[k]}$, the scaled magnitude of the same vectors, has pdfs corresponding to the *Rayleigh* and *Rician* distributions:

$$\begin{aligned} &f_0\left(\sqrt{S\left[k\right]}\right) \text{ is a Rayleigh distribution with scale parameter } \sigma = 1, \\ &f_1\left(\sqrt{S\left[k\right]}\right) \text{ is a Rician distribution with scale parameter } \sigma = 1 \\ &\qquad \text{and distance parameter } \nu = \sqrt{\delta} = 2\left(\frac{|m|}{\sqrt{K}\sigma_w}\right). \end{aligned} \tag{4.28}$$

Now that the requisite pdfs have been defined, the performance of this occupancy detector as implied by (4.24) was assessed in a similar fashion to the peak detection techniques described previously in this chapter. As an implementation detail, *log-likelihoods* were used, converting (4.24) into:

$$y = \underset{y\in[1\ldots Y]}{\arg\max}\left(\sum_{k=1}^{K}\log\left(f_{O_y[k],k}\left(S[k]\right)\right)\right). \tag{4.29}$$

TABLE 4.2: Detection Simulation using Bayesian Decisions, When Licensed Network User Signal is Present

		Simulation Cases			
		No offset, SNR = 0 dB	**+1024 offset, SNR = 0 dB**	**-1024 offset, SNR = 0 dB**	**No offset, SNR = -3 dB**
Parameters	**No. detection events**	15000			
	Noise	Additive White Gaussian Noise (AWGN)			
	Licensed network user signal	OFDM with symbol length 256; data sub-carriers QPSK modulated; no guard interval; no pilot subcarriers			
	No. freq. bins (K)	4096	4096	4096	4096
	Freq. occupancy $A_L \dots A_U$	1921 ... 2176	2945 ... 3200	897 ... 1152	1921 ... 2976
	SNR (dB) (in occupancy bins)	0	0	0	-3
	Missed detection rate P_{md}[1]	0.01			
Results	**Probability of correct D_L ($f_L[0]$)**	0.188	0.184	0.183	0.069
	Probability of correct D_U ($f_U[0]$)	0.188	0.189	0.183	0.076
	D_L bias $(\overline{D}_L - A_L)^2$	1.0	0.8	1.0	2.9
	D_U bias $(\overline{D}_U - A_U)^2$	-0.9	-0.9	-0.9	-1.7
	Spread s_{D_L}	5	5	5	16
	Spread s_{D_U}	5	5	5	17
	Perfectly correct (%)[3]	3.93	3.46	3.85	0.55
	Completely incorrect (%)[3]	0.00	0.00	0.00	0.04
	False-alarm spectral set (%)[3]	55.6	57.3	56.0	63.8
	Mean false alarm spect. set size[4]	8	8	8	23
	Miss.-detect. spectral set (%)[3]	72.3	72.1	72.6	77.6
	Mean miss. det. spect. set size[4]	9	9	10	26

[1] The detection threshold λ_T was set such that the proportion of detection events where the test statistic λ was less than this (resulting in an incorrect H_0 decision), equalled P_{md}.
[2] Here, *bias* is the difference between the *sample mean* of the estimated value, calculated over all detection events, and the actual value..
[3] These percentages are calculated from those detection events where the test statistic λ was above the detection threshold λ_T (resulting in H_1 decisions). Also refer to section 4.1.3.
[4] *Size* refers to the number of frequency bins in $\mathbf{S_{fa}}$ or $\mathbf{S_{md}}$ — refer to section 4.1.3.

This formulation avoids some potential precision issues relating to arithmetic *underflow* when multiplying many very small numbers together – something that is worth considering in real SDR implementations that may be using fixed-point or limited-precision floating-point arithmetic. Of course, this comes at a cost of requiring additional logarithms to be calculated. Alternatively, the logarithm of the pdf values can be *pre-calculated*.

Simulation parameters and results have been summarized in TABLE 4.2, and the pmfs of the boundary decisions are shown in FIGURE 4.6.

Direct comparison of these results with those from the maximum-slope-magnitude peak detection described in section 4.2.1 is problematic, as the nature of the licensed network user's transmission is slightly different in both simulations and the beneficial

effect of the edge-aligned pilot sub-carriers in aiding detection is absent in this simulation (the simulation case most similar to that used here would be *No offset, no pilots*) – however, some results are apparent:

- The proportion of accurate estimates (where $D_L = A_L$ and $D_U = A_U$) is higher for the *-3* dB SNR case here compared to the *-3* dB SNR case in the maximum-slope-boundary peak detection simulation, and the same applies to the *0* dB SNR case here compared to the *No offset, no pilots* case which also had *0* dB SNR (a desirable outcome)

- The spread of the estimates, s_{D_L} and s_{D_U}, are lower for the *0* dB SNR and *-3* dB SNR cases here, compared to the cases with corresponding SNR values in the maximum-slope simulation (which is desirable)

- More *perfectly correct* detections (explained in section 4.1.3) occur in the *0* dB SNR cases here, compared to the *No offset, no pilots* case of the maximum-slope simulation (which is desirable)

- Fewer *completely incorrect* detections (explained in section 4.1.3) occur in the *-3* dB SNR case here, compared to the *SNR = -3* dB case in the maximum-slope simulation (which is desirable)

- The bias in D_L and D_U seen here is generally symmetrical in magnitude (but opposite in sign) – similar to the *No offset, no pilots* case in the maximum-slope simulation

- Where missed detections occur on one or more frequency bins, the average number of occupied bins missed (*mean missed-detection spectral set size*) is fewer in these cases compared to all cases in the maximum-slope simulation with the same SNR – in other words, when the licensed network user is transmitting, fewer of the spectral components of its signal is missed by the detector (which is desirable)

- The pmfs of D_L and D_U here appear to be symmetrical – similar to the *No offset, no pilots* case in the maximum-slope boundary simulation (where there are also

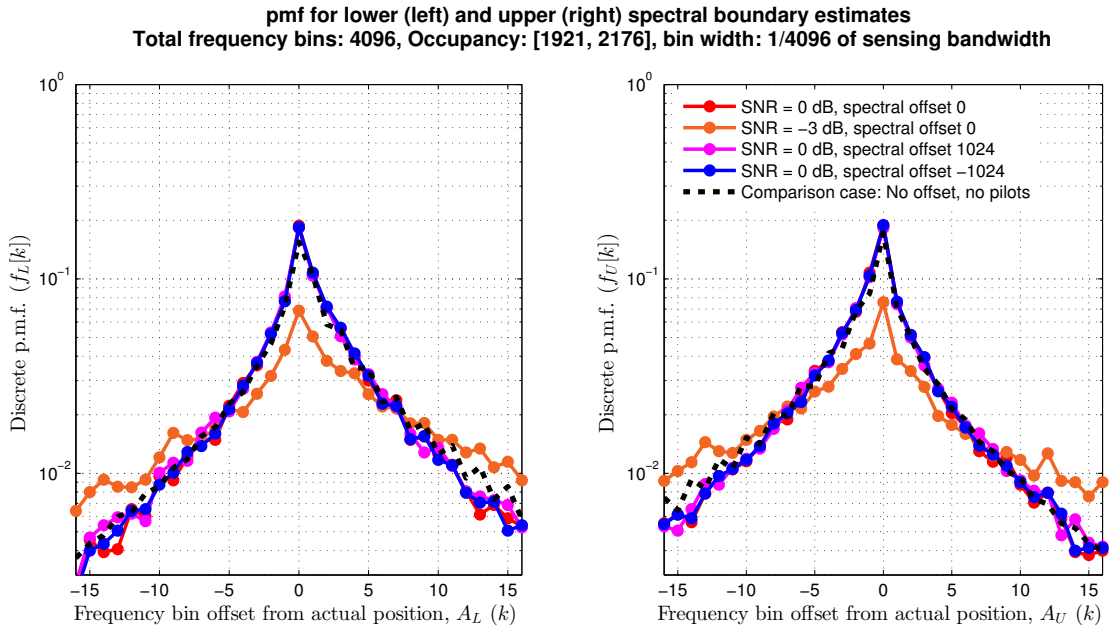

FIGURE 4.6: pmfs for the lower and upper spectral boundary estimates using Bayesian peak detection. The result from the *No offset, no pilots* case in TABLE 4.1 has also been superimposed for comparison

no edge-aligned pilot sub-carriers)

It should be noted at this point that this approach is very computationally complex, and relies on prior knowledge of probability distributions that a wireless device is unlikely to have in a practical system. Further, the simplifying assumptions make the detection scenario described unrealistic. These results may be considered to be a reference implementation whose performance reflects the exploitation of detailed knowledge about the transmitted signal of a licensed network users within the radio environment.

4.4 Conclusion

The traditional energy detection concept has been extended so that it not only makes a decision about whether or not a signal is present, but also its spectral occupancy within a sensing band. A low-complexity peak detection algorithm, needed to provide this additional capability, was described, and its performance demonstrated under several

simulated radio environments. This extended detection model is more suitable for truly cognitive radios performing dynamic spectrum access in environments subject to arbitrary signal channelizations, where incumbent licensed network users may not transmit according to centrally-managed or conventionally agreed-upon channels.

The peak detection approach described in section 4.2 is relatively simple to implement, and shares many similarities to a frequency-domain implementation of a conventional energy detector. However, the estimators for the occupancy boundaries have a bias that appears to be dependent on the spectral offset of the signal within the sensing bandwidth. This is not ideal, and removing or at least equalizing the bias could motivate further investigation. The approach was compared to a technique based on Bayesian decision theory. Whilst improved spectral boundary detection outcomes were apparent with this approach, it is of theoretical interest only as a reference, as it is extremely computationally demanding and requires prior knowledge about the precise spectral distribution of the power in the signal of interest – information that a cognitive radio terminal is unlikely to have. The performance of the Bayesian approach could be modified by removing the assumption of equal a-priori probability associated with each hypothesis in (4.14). If the the a-priori probabilities are different and known, then the calculation described by (4.18) turns the decision process from maximum-likelihood to *a-posteriori* in nature. If it were possible to use Bayesian detection in a cognitive radio terminal, these a-priori probabilities could be estimated from a database of prior decisions – it could effectively refine its decision process by 'learning from experience'.

The verification of the peak detection schemes took place with a PSD estimate of synthesized wideband signals. It would be interesting to see how well the detector works with signals from real-world radio environment measurements. The PSD-estimation approach was based on a standard *periodogram*. Other PSD-estimation approaches such as *Welch's method* may result in a higher-quality and smoother PSD estimate but at a cost – either the spectral resolution must be reduced, or a longer signal observation must be taken to maintain the same spectral resolution.

The spectrum is taken from the FFT of the observation, and hence the spectral resolution of the detection process is limited by the number of signal samples taken

in the observation – the resolution may be increased at the cost of additional computation needed to analyze and process longer signal observations (by calculating a larger FFT). Alternative spectral-analysis techniques such as *wavelet transforms* (used in, for example, [81], [82], [83]) may overcome this concern to some extent – being a *multi-resolution* transform, it allows for a more flexible trade-off between spectral and temporal resolution. A truly cognitive radio may find this extra degree of freedom useful, although these approaches also required large computational capability. It is also conceivable that edge-detection methods borrowed from the field of image processing might prove to be useful. It would be insightful to consider further the application of these techniques to the same problem in later work.

It would be useful to consider a metric that allows a direct performance comparison to be made between the extended detector model and a traditional energy detector. Conceptually, one expects that the extended detector must have worse performance, as the occupancy decision is now subject to additional stochastic variability via the spectral boundary estimates. The extended detector's concept of false alarm and missed detection is somewhat different to that which applies to traditional energy detection, due to the extra dimensions in the decision that is made. These two sets of concepts require reconciliation before meaningful comparison can be made.

Finally, just like co-operative spectrum sensing approaches have been shown to improve traditional signal detection results by 'fusing' the decisions from multiple terminals, one would expect that co-operative approaches would lead to improved decisions in this application too.

5

Optimize: Selecting from Many Available Vacant Channels

Dynamic spectrum access by cognitive wireless devices requires the ability for the terminal to make an assessment of its radio environment, and then identify what channels it could make use of without interfering with any existing devices with licensed access to the spectrum. As indicated in section 1.1.3, opportunities which can be exploited in this manner can vary along several dimensions such as frequency (the spectral dimension) and time (the temporal dimension).

In Chapters 2 to 3, the emphasis was on improving the capacity to determine correctly if a particular channel was occupied or not, *at a given point in time*. In Chapter 4, the emphasis was on identifying the spectral locations of licensed network users within a wide sensing band, unencumbered by any presuppositions about the

centre frequency and bandwidth of such devices.

Determining if a channel is occupied or not is important – but it only tells a part of the story. For example, say a device determines that two channels are vacant – how does it select which one it should use? The 'unoccupied/occupied' decision, by itself, doesn't provide any useful basis upon which to make this selection, and it is unlikely that simply picking one of the options at random will yield an optimal outcome. To quote from section 1.1.3, '*not all spectral opportunities are created equal*'.

This chapter looks at additional attributes of a channel that provide information relating to the temporal characteristics of its occupancy. The aim is to make better use of the temporal information that is observed about a channel, and illustrate how this can be incorporated into the cognition and decision-making cycle. This paves the way for dynamic spectrum access that is '*learning enhanced*' [42].

In this chapter:

- Some statistics and quantifiable attributes, relevant to characterizing the temporal character of the occupancy of a channel, are presented

- Measures for reducing the computational complexity associated with obtaining these attributes are presented (as they may need to be frequently evaluated to cater for changing radio environments)

- It is shown that machine learning algorithms can be used to derive a *channel selection policy*, which employs these channel attributes, so that a cognitive radio can optimize an overall objective based upon balancing its own throughput against exerting interference on licensed network users

- The efficacy of such channel selection policies is evaluated through simulations, incorporating radio environments whose spectral and temporal characteristics are based on a corpus of real-life observations

5.1 The Rationale for Distinguishing Between Different Channel Occupancies

During 'radio scene analysis' (see section 1.1.2), a cognitive radio performing dynamic spectrum access is faced with a dilemma: *given a choice of several vacant channels to pick from, which one should it select for use?* Channels which are occasionally occupied by a licensed network user, for example, will exhibit different characteristics over time that affect the *utility* that a cognitive radio would expect from utilizing one for its own communications needs, compared to one that is occupied nearly all the time. For example, a cognitive radio may see that a channel is vacant (momentarily) – however, that channel is in fact, most of the time, occupied by a licensed network user. In fact, the utility obtained by using that channel will likely be short-lived, as the cognitive radio will probably only be able to transmit a small amount of data before the channel becomes occupied again and it is forced to vacate it. Doing this incurs costs associated with network re-establishment, channel switching, traffic delay, and some period of interfering with the licensed network user, which may nullify the small expected benefit. On the other hand, it is plausible that a vacant channel that history shows is only occupied sporadically is a much better candidate for exploitation, as the expected costs associated with using it are lower and will be incurred much less frequently.

Furthermore, some channels may be occupied according to a fairly predictable pattern, e.g. the licensed network user may only transmit at certain times of the day, or it may transmit with a consistent burst length and/or consistent interval between bursts. It is reasonable to expect that more benefit could be derived from exploiting such channels, compared to ones whose occupancy pattern is random and unpredictable. For example, it would be possible to predict with greater success when the licensed network user is going to become active, and even take pre-emptive measures to vacate the channel so as minimize interference. This suggests a role for *machine learning* to analyze the historical occupancy of channels and recognize the presence of recurring or predictable occupancy patterns.

In relation to a channel that appears to be vacant, a more fundamental question could be asked: *is it even worth attempting to use that channel?* The example above where a channel is occupied by a licensed network user most of the time is one scenario where this question is particularly relevant. If the expected costs of using the channel outweigh the benefits, resulting in essentially negative utility, it may actually be better to stop transmitting and enter a quiescent state whilst waiting for a better opportunity to emerge, rather than attempting to exploit the channel.

A cognitive radio, as an agent endowed with some level of machine intelligence, should seek to incorporate as much information about the channels it has to select from, in order to attempt to answer the above questions, and hence make spectrum access decisions that are (sufficiently close to being) optimal. Therefore, this motivates a more detailed characterization of exploitable channels, beyond simply whether or not they are occupied at a given time. A cognitive radio can then learn, from experience, the expected utility that would be derived from exploiting opportunities with different characterizations, and hence formulate some *channel selection policy* to adhere to during operation.

5.2 Characterizing Channel Occupancy Patterns

A radio environment will consist of some combination of white, grey and black spaces (refer to section 1.1.3). White spaces are of most value to a cognitive radio performing dynamic spectrum access, as such a channel can be used without any possible repercussion associated with exerting interference on a licensed network user. However, depending on the radio environment and how 'crowded' it is, white spaces may be difficult to find, or not exist at all. Black spaces, if recognized, provide no resource that is exploitable by the cognitive radio and can be ignored. It is the grey spaces in between that will likely present the most spectral opportunity – their characterizations will vary along some continuum between that of white and black spaces.

Referring back to (2.2), it can be seen that the occupancy of any channel of arbitrary bandwidth over time can be modelled as a sequence of random variables R, hence

making it a *binary* sequence. Formalizing this:

$$\mathbf{o} = [o[1]\, o[2]\, o[3]\, \ldots\, o[N]]\,, \tag{5.1}$$

where $o[n]$ is the occupancy status of a given channel at a time index n:

$$\begin{aligned} &o[n] \in \{0, 1\}\,, \\ &o[n] = \begin{cases} 0 & \text{when the channel is vacant at discrete time } n \\ 1 & \text{when the channel is occupied at discrete time } n, \end{cases} \end{aligned} \tag{5.2}$$

and N is the length of the occupancy sequence. Note that the entries $o[n]$ can be either based on the 'reality' of the occupancy (see (2.2)), or the corresponding decisions resulting from a signal detection algorithm (see (2.7)), depending on context.

What qualities can be called upon to characterize such occupancy sequences? In the work presented in [42], two attributes are used, namely *duty cycle* and *complexity*, which are elaborated upon below.

5.2.1 Duty Cycle

The duty cycle is simply the overall proportion of time when the channel is occupied. In other words, it is the probability that a channel is occupied in a given time interval. From an occupancy sequence, it is estimated by the fraction of entries in $\mathbf{o}$ that are equal to 1.

5.2.2 Unpredictability of the Occupancy Pattern

In this context, the term *complexity* refers to a measure of the amount of 'randomness' or unpredictability associated with the terms in the occupancy sequence – sequences that are highly complex are very random and unpredictable. The measure of complexity used in [42] is the Lempel-Ziv complexity (LZC), although other measures are possible. LZC calculation does not require any prior assumption to be made about the random process that generates the occupancy sequence [42]. In [43], the authors

present complexity as a measure of how much a sequence 'resembles' one where every possible unique value occurs with equal probability. This definition makes complexity very much related to the term *entropy* as it applies to information theory. Where each value in an occupancy sequence is considered to be an instance of a random variable, the entropy measures the uncertainty of the random variable. For a binary random variable such as occupancy, its entropy is defined as [84]:

$$H(O) = -\sum_{k=0}^{1} P\left(O = k\right) \log_2 \left(P(O = k)\right), \tag{5.3}$$

where O represents the occupancy random variable, and $P\left(O = k\right)$ is the probability that it has a value k. Like complexity, entropy is maximized when the random variable takes on every possible value with equal probability [84]. By substituting $P\left(O = 0\right) = P\left(O = 1\right) = 0.5$ into (5.3), it can be seen that *the maximal entropy of a truly random binary occupancy sequence is equal to 1.*

The calculation of LZC is described in detail in [43] – however, some main ideas will be reiterated here.

The foundation of the calculation comes from the determination of a sequence's *eigenfunction*, which is a discrete, *monotonically increasing* function of the sequence's index. The eigenfunction of a sequence indicates its rate of *innovation*, with its value indicating the size of an implied *cumulative* vocabulary as the sequence index increases. Highly-complex sequences are therefore also highly innovative, and complexity is then a measure of the average rate of increase of the sequence's eigenfunction.

Each term (or *symbol*) in the sequence is taken from an *alphabet*, A, of length α. In this case, the occupancy sequences are binary, so:

$$\begin{aligned} A &= \{0, 1\}, \\ \alpha &= |A| = 2. \end{aligned} \tag{5.4}$$

Any *sub-sequence* of the sequence can be defined by two integer indices, i and j,

which specify where in the original sequence the sub-sequence starts and finishes, as follows (referring to (5.1)):

$$\mathbf{o}_{i,j} = [[o[i]\ o[i+1]\ o[i+2]\ \ldots\ o[j-1]\ o[j]]\,. \tag{5.5}$$

When the start and end positions are the same, the result is a sequence of length 1:

$$\mathbf{o}_{j,j} = o[j]. \tag{5.6}$$

Furthermore, placing two sequences next to each other without spaces results in a new sequence that is a *concatenation* of the two:

$$\mathbf{o}_{i,k} = \mathbf{o}_{i,j}\mathbf{o}_{j,k}. \tag{5.7}$$

A sequence's *vocabulary*, V, is the set of all unique sub-sequences of a sequence (*including the sequence itself*). By definition, the empty sub-sequence, $\varnothing$, which occurs when $j < i$, is considered to be part of the vocabulary. A *proper prefix* of a sequence is any sub-sequence with $i = 1$ and $j < N$, N being the length of the sequence. On the other hand, a *suffix* of a sequence is any sub-sequence with $1 \geq i \geq N$ and $j = N$. The *eigenvocabulary* of a sequence, E, is the set of sub-sequences which do not belong to the vocabulary of any proper prefix of the sequence, and the elements of the eigenvocabulary are called *eigenwords*. The cardinality of the eigenvocabulary of a sequence is known as its *eigenvalue*.

To illustrate, consider an occupancy sequence given by $\mathbf{o} = [010001]$. In this case, it can be seen that:

- $V(\mathbf{o}) = \{\varnothing, 0, 1, 01, 10, 00, 010, 100, 000, 001, 0100, 1000, 0001, 01000, 10001, 010001\}$
- List of proper prefixes: $\{\varnothing, 0, 01, 010, 0100, 01000\}$; each of these have their own vocabularies
- $E(\mathbf{o}) = \{001, 0001, 10001, 010001\} \subset V(\mathbf{o})$
- $|E(\mathbf{o})| = 4$ is the eigenvalue of $\mathbf{o}$

The sequence of eigenvalues for all sub-sequences $\mathbf{o}_{1,j}$, ordered by increasing j, where $1 \leq j \leq |\mathbf{o}|$, is called the *eigenfunction* of $\mathbf{o}$. It can be represented by a discrete function, $f_{\mathbf{o}}^{\lambda}[k]$:

$$f_{\mathbf{o}}^{\lambda}[k] = \left|E\left(\mathbf{o}_{1,k}\right)\right|, \; 1 \leq k \leq N. \tag{5.8}$$

It can be shown that the eigenfunction of a sequence is non-decreasing [43].

A sequence can be decomposed into a *production history*, $H\left(\mathbf{o}\right)$, of length m, where each component is a sub-sequence of the original sequence, and the original sequence can be reconstituted by concatenating together the elements of the production history:

$$H\left(\mathbf{o}\right) = \left\{\mathbf{o}_{1,k_1}, \mathbf{o}_{k_1+1,k_2}, \mathbf{o}_{k_2+1,k_3}, \ldots, \mathbf{o}_{k_{m-1}+1,k_m}\right\}, \; k_m = N. \tag{5.9}$$

The sequence of indices $k_1, k_2, k_3, \ldots, k_m$ defines the 'parsing points' for the production history. The production history implies a *complexity* value for the sequence, given by the length of the history:

$$C\left(\mathbf{o}\right) = \left|H\left(\mathbf{o}\right)\right| = m. \tag{5.10}$$

In [43], two different rules are specified for determining these parsing points.

One rule, which decomposes a sequence into an *exhaustive history*, $\left(H_e\left(\mathbf{o}\right)\right)$, defines the parsing points as follows [43]:

$$H_e\left(\mathbf{o}\right) = \left\{\mathbf{o}_{k_{i-1}+1,k_i}\right\}, \text{ where } k_i = \min\left\{k \middle| \left(f_{\mathbf{o}}^{\lambda}\left[k\right] > k_{i-1}\right)\right\} \text{ for } i < m$$
$$k_0 = 0, k_m = N. \tag{5.11}$$

In words, a parsing point is the *minimum* sequence index such that the corresponding eigenfunction value is greater than the previous parsing point, with the initial parsing point defined at index 0, except for the final parsing point which is always at the end of the sequence. The size of the exhaustive history is then defined as the *exhaustive*

complexity of the sequence:

$$C_e(\mathbf{o}) = |H_e(\mathbf{o})|. \tag{5.12}$$

Another rule decomposes a sequence into a *primitive history*, with the parsing points defined as follows [43]:

$$H_p(\mathbf{o}) = \left\{\mathbf{o}_{k_{i-1}+1,k_i}\right\}, \text{ where } k_i = \min\left\{k \middle| \left(f_{\mathbf{o}}^{\lambda}[k] > f_{\mathbf{o}}^{\lambda}[k_{i-1}]\right)\right\} \text{ for } i < m$$
$$k_0 = 0, k_m = N, f_{\mathbf{o}}^{\lambda}[0] = 0. \tag{5.13}$$

In words, a parsing point is the *minimum* sequence index such that the corresponding eigenfunction value is greater than the previous parsing point's eigenfunction value, with the initial parsing point defined at index 0 and its corresponding eigenfunction value also defined to be 0. The final parsing point is always at the end of the sequence. The size of the primitive history is then defined as the *primitive complexity* of the sequence:

$$C_p(\mathbf{o}) = |H_p(\mathbf{o})|. \tag{5.14}$$

According to [43], the complexity of a sequence is lower-bounded by its primitive history, and upper-bounded by its exhaustive history.

It is useful to *normalize* the raw complexity values, taking into account the length of the sequence involved. In this work, normalized complexity is defined as follows:

$$\hat{C}(\mathbf{o}) = \frac{C(\mathbf{o})}{\left(\frac{N}{\log_\alpha(N)}\right)}, \alpha = 2 \text{ (being a binary sequence)}, \tag{5.15}$$

where $C(\mathbf{o})$ is a measure of the complexity of the occupancy sequence, and N is the sequence's length. The normalization term $\left(\frac{N}{\log_\alpha(N)}\right)$ is based on the relationship between the exhaustive complexity of a sequence and the *entropy* of the terms in that sequence when considered as a random variable. Both complexity and entropy

play a similar role in characterizing the 'unpredictability' associated with an sequence, and the entropy of the values making up the sequence is equal to the expectation of the sequence's complexity [85]. In [43], it is additionally stated that the exhaustive complexity of a binary sequence can be asymptotically upper-bounded as follows:

$$C(\mathbf{o}) \leq \frac{HN}{\log_\alpha(N)} \text{ as } N \to \infty, \tag{5.16}$$

where H is the entropy of the sequence terms as defined in (5.3), and the random process which generates the sequence is considered *ergodic*. Therefore, the normalization term is the asymptotic upper bound of the exhaustive complexity of a 'purely random' binary sequence generator which has $H = 1$ (as indicated by (5.3)). This normalization is also used in [86].

In subsequent references to LZC, unless stated otherwise, *normalized exhaustive complexity* is implied – as defined by (5.12) and (5.15).

5.2.3 The Relationship between Duty Cycle and LZ Complexity

Channels that are entirely vacant, or fully occupied, have well-defined characterizations in terms of DC and LZC:

- Completely vacant channels can be represented by a sequence consisting only of 0s. DC = 0, LZC $\to$ 0[1].
- Fully-occupied channels can be represented by a sequence consisting only of 1s. DC = 1, LZC $\to$ 0.

However, the 'grey spaces' between these two extremities (see section 1.1.3) may exhibit a wide range of DC and LZC values. FIGURE 5.1 illustrates the many-to-many relationship between DC and LZC, particularly around the central regions of the scatter plot corresponding with intermediate values of DC and LZC. This scatter plot represents DC and LZC measures for occupancy sequences for a channel in the 770 MHz

[1]Unnormalized LZC is exactly 1, so normalized LZC approaches 0 asymptotically as the length of the sequence increases.

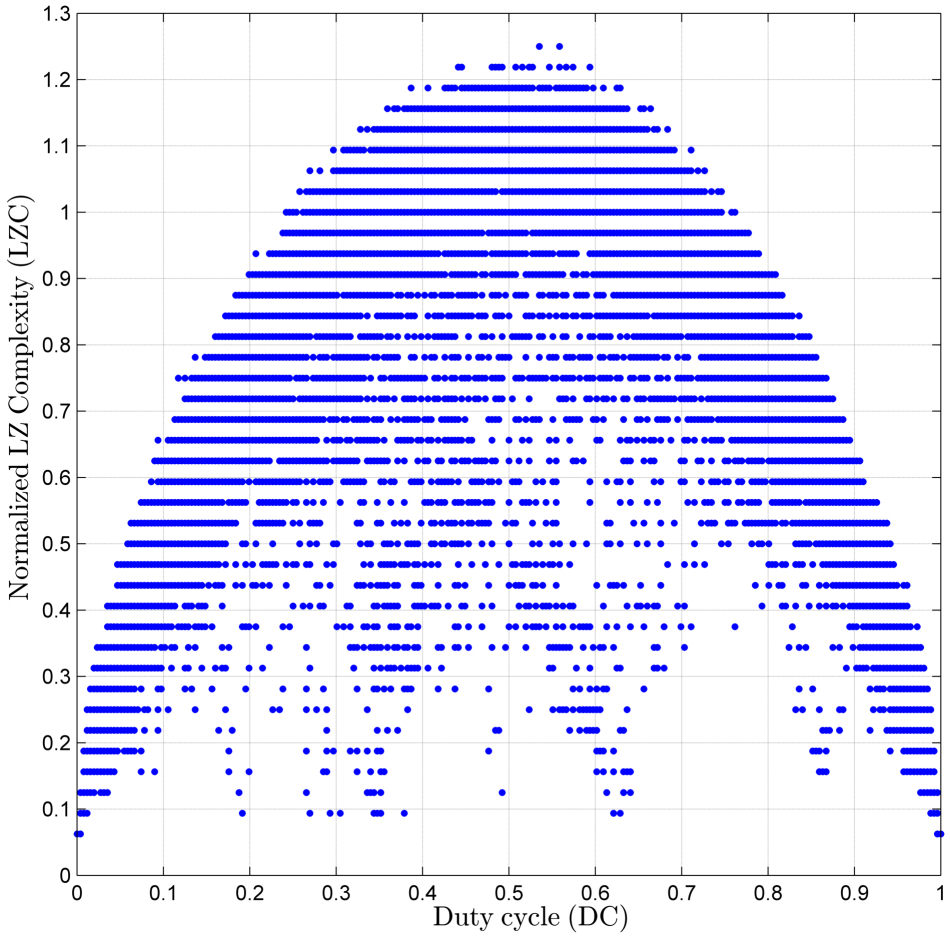

Figure 5.1: Scatter plot showing some of the DC and LZC values associated with occupancy sequences obtained from a spectrum occupancy measurement campaign conducted by RWTH Aachen University in Aachen, Germany. DC and LZC values were calculated from occupancy sequences of length $N = 256$, obtained from measurements taken indoors inside an office building.

region, as measured in an indoor office building in Aachen, Germany, in January 2007 [87]. The data set relates to channels of 200 kHz bandwidth, within a frequency band spanning 1500 MHz from 20 MHz to 1520 MHz. A received power measurement was taken in each 200 kHz channel once per second, and it was deemed to be occupied if the received power was greater than -107 dBm.

The independence between DC and LZC is also illustrated in Figure 5.2. Qualitatively, even low-DC sequences can exhibit relatively high LZC, depending on how unpredictable and random are the lengths of time when the channel is unoccupied (represented by sub-sequences of 0s). Similarly, even high-DC sequences can exhibit

FIGURE 5.2: For occupancy sequences, there is a many-to-many relationship between DC and LZC.

relatively high LZC, depending on how unpredictable and random are the lengths of time when the channel is occupied (represented by sub-sequences of 1s). For medium-DC sequences, the LZC varies depending on how unpredictable and random are the pattern of 0s and 1s.

5.3 Learning a Channel Selection Policy

Now that the way in which channels could be characterized has been established, attention must turn to how this information can be used to formulate a channel selection policy.

In this context, *a channel selection policy refers to a procedure, given a condition where a cognitive radio performing dynamic spectrum access is required to vacate the channel that it is currently utilizing, for determining what channel it should select to switch to, from the available options.*

This implies some form of machine intelligence and, in the absence of past experience from which to draw upon immediately, machine learning. At this point, cognitive abilities, in the true sense of the word, start to apply.

Given the heterogeneity of radio environments in which a mobile cognitive handset might find itself, machine learning will be a crucial building block in building up this capability, as it will allow the handset to 'learn from experience' obtained from the environment, rather than simply following some generic set of rules which have been hard-coded in advance. Reinforcement learning as a general framework is suitable for

this purpose.

5.3.1 Reinforcement Learning and Decision Processes

In broad terms, reinforcement learning is an approach whereby a learning *agent* interacts with its *environment* by taking some *action*, and in response, the environment presents a modified environment, as well as some *reward* [88]. A reward is an abstract concept, expressed numerically, which represents some outcome that the cognitive radio tries to maximize over time by the actions that it chooses to take. Rewards can be signed numbers, such that negative values can indicate conceptual *penalties* or *costs*. The concept of reward in this context is closely related to that of *utility*, the result of a function that must be maximized in an optimization problem. Conceptually, the formulation of a policy, which guides the selection of actions towards maximizing the reward obtained from the environment over time, can be seen as the solution to an optimization problem. Hereafter, the term 'utility' is used used to mean the same as 'reward' (benefits minus costs). The rewards concept allows the agent to 'learn from experience' – in a particular situation presented by the environment, past actions in the same situation which have yielded less reward can be spurned in favour of more rewarding actions.

In this context:

- The learning agent is the cognitive radio performing dynamic spectrum access
- The environment is the radio environment, consisting of spectral bands or channels which are exploitable by the cognitive radio, and the activity of licensed network users transmitting for some of the time on these channels
- Utility is defined by the benefits associated with transmitting, such as the value of the data transferred over a cognitive radio network from a transmitter to a receiver, minus the costs and penalties associated with doing so e.g. interference exerted upon a licensed network user's transmission, network latency associated with switching channels, battery power consumed.

Generally, the interactions between the agent and its environment (when decisions about actions get made) take place at discrete times, or *epochs*. They may or may not be evenly spaced in real time – they are simply successive stages whereby the agent is able to initiate a new action [88]. At each epoch, the environment may change. At this point, there is a notable parallel between the discrete times where the agent is able to take actions, and the discrete times at which the occupancy of a channel can be evaluated, suggesting how the latter can be incorporated into a reinforcement learning structure. Also, the information that defines the environment may be considered collectively as its *state*.

In many machine learning problems, the agent is seeking to fulfill a specific goal or *objective* which is clearly defined. A state which corresponds with an objective being achieved is called a *terminal state* or *absorbing state*. In such problems, the concept of utility needs to be framed so as to guide the agent towards achieving the objective. In other words, by learning to maximize its utility over time, the agent will then naturally learn how to achieve its objective – the two concepts are intricately related.

Some applications have a natural concept of a terminal state, some do not. In the case of a cognitive radio which has an unending stream of data to transmit over time, such an application would fall into the latter category. However, even with a lack of a terminal state, the agent can still learn an action policy that yields the maximum utility over time.

Whilst the agent is going through its learning phase, it is learning the action-selection policy that will maximize its utility over time. In the absence of any prior knowledge from which to guide the selection of actions, the agent must undertake *exploration*, selecting actions stochastically. The agent may be placed in an initial, randomly selected state. It then proceeds, at each epoch, to take an action in response to the new state presented to it, whilst accumulating information about the utility obtained from taking these actions. For problems where a terminal state exists, a learning *episode* is completed when this is reached. A new episode can then begin in the same way, each episode further refining the cumulative utility information gained. For problems where there is no terminal state, there is only one learning episode, which

never ends. However, learning about the utility gained by taking certain actions at each epoch can still take place.

During learning, actions are selected in response to the prevailing state according to a *policy*. This policy must ensure that sufficient coverage of the entire action space takes place. This is particularly important for problems without a terminal state and distinct learning episodes, where a new episode may begin in a randomly selected state. Coverage is important, otherwise it may be possible that one of the actions which are not covered in a given state is, in fact, the optimal one. Therefore, the policy used in learning and exploration is stochastic in nature, and is referred to as the *behaviour policy* [88]. During learning and exploration, the policy used to select actions may not actually be the one that is optimal. *The optimal policy is deterministic rather than stochastic.* A reinforcement learning algorithm must attempt to estimate this optimal policy, and *this is the one that is used by the agent once learning has been completed* – this is the *estimation policy* [88]. A 'channel selection policy' used by a cognitive radio in operation is an example of an estimation policy. Reinforcement learning algorithms may be classified according to how these two types of policies are related [88]:

- **On-policy learning**: The behaviour policy is iteratively evaluated and then modified in an attempt to make it approach the optimal estimation policy. An iteration may happen after each learning episode or at each epoch.

- **Off-policy learning**: The behaviour estimation policies are decoupled from each other, so that the estimation policy can be improved with each learning episode or epoch, independently of the behaviour policy used.

Off-policy approaches to learning have a powerful advantage – they allow the use of a highly exploratory behaviour policy that is *not necessarily the optimal policy*, whilst independently of this, the estimation policy still converges towards an optimal one. For example, the behaviour policy could simply be to select the next action at random – an estimate of the optimal policy still emerges once learning is completed. *Q-learning* is an example of an off-policy approach, which is described in more detail later.

As the environment can be modelled by some state information which changes from one epoch to the next, this can be considered a *state transition*. Furthermore, given a current state, the next state is determined, both by an underlying stochastic process, as well as the action that is exerted by the agent. When there is a dependence on both a random component (outside the control of the agent) and a controlled component (the chosen action of the agent) in the environmental state transitions, the decision-making process of the agent can be modelled as a *Markov decision process* (MDP). This dual dependence is illustrated by the notation used to describe MDPs, such as in [88].

Suppose state is denoted by the symbol s, which is a member of the set of all possible states, $\mathcal{S}$:

$$s \in \mathcal{S}. \tag{5.17}$$

Let the action taken be denoted by a, which is a member of $\mathcal{A}_s$, the set of all possible actions given the current state is s:

$$a \in \mathcal{A}_s. \tag{5.18}$$

In situations where the available actions are always the same, regardless of the state s, the subscript s may be dropped. Both states and actions can be indexed by a discrete-time epoch t which indicates when the state was observed or an action was taken by the agent, $t = 0, 1, \ldots, \infty$. Under an MDP, the *state transition probabilities* for a state s observed at time t, and a state s' at time $t + 1$, is given by:

$$P_a(s, s') = P(s_{t+1} = s' | s_t = s, a_t = a). \tag{5.19}$$

Such a decision process is denoted as 'Markov' because the environment satisfies the *Markov property*, whereby the next state is only explicitly dependent on the present state and action, and not on the prior history of states and actions[2]. In other words,

[2]This may be described explicitly as a *first-order* Markov process, with the order indicating the length of the history of states and actions that the determination of the next state explicitly relies upon.

$P_a(s, s')$ can be evaluated independently of the values of states $s_{t-1}, a_{t-1}, s_{t-2}, a_{t-2}, \ldots, s_0, a_0$.

More generally for reinforcement learning, a policy specifies the probability of taking an action a when in state s:

$$\pi(s, a) = P(a|s). \tag{5.20}$$

A deterministic policy, such as one used once learning is complete, is one where:

$$\exists a : P(a|s) = 1. \tag{5.21}$$

Previously, it has been stated that reinforcement learning attempts to determine a policy that will result in maximal utility being received to the agent over time – this needs to be defined explicitly. In [88], the value to be maximized is called the *return*. At each epoch, some *immediate* utility is received by the agent from its environment, as a result of taking some action at the previous epoch. The return must be evaluated going forward from some time t, and is a function of the sequence of immediate utilities received at each time epoch subsequent to t. Let the immediate utility received at epoch t be called r_t, and the return *subsequent to* t be R_t. If R_t was to be evaluated by a simple summation of all the individual rewards $r_t, r_{t+1}, r_{t+2}, \ldots$, then a mathematical dilemma results for problems where there is no terminal state, as it may be possible that $R_t = \infty$. Therefore, reinforcement learning generally attempts to maximize a *discounted return*, defined as follows [88]:

$$R_t = \sum_{k=1}^{T} \gamma^{k-1} r_{t+k}, \text{ with } 0 < \gamma < 1, \tag{5.22}$$

where γ is the *discount factor*, and T is the number of additional epochs needed to reach the terminal state from t. Where no terminal state exists, $T = \infty$, but the summation will still converge as long as the individual utility values are bounded, due to the restriction on the range of γ values. The learning result is influenced by the selection of γ. When $\gamma \to 0^+$, the objective is to only maximize immediate rewards, as utilities in subsequent epochs are always zero, leading to *myopic* behaviour. On the

other hand, as $\gamma \to 1^{-}$, an increasing emphasis is placed on future utility in the overall reward, and the estimation policy strives for *long-term* utility during learning.

For reinforcement learning generally, the optimality of any policy is evaluated by considering either the resulting *state-value function* or *action-value function*. The state-value function basically assigns a value to each state which indicates how 'desirable' that state is, based on expected return from that state, and is policy-dependent [88]:

$$V_{\pi}(s) = E\{R_t|\pi, s_t = s\}, \tag{5.23}$$

where $E\{\bullet\}$ is the expectation operator, and the expectation is evaluated, for a fixed policy π, over all possible sequences of state-action pairs that can occur with s_t as the starting point, until the terminal state is reached. Similarly, the action-value function $Q_{\pi}(s, a)$ indicates how desirable taking a certain action a is, in response to the environment presenting a state s, and is again policy-dependent [88]. It may be defined as:

$$Q_{\pi}(s, a) = E\{R_t|\pi, s_t = s, a_t = a\}. \tag{5.24}$$

Again, the expectation is evaluated, for a fixed policy π, over all possible sequences of state-action pairs that can occur with s_t as the starting point, until the terminal state is reached. It gives the expected return from taking action a at time t whilst in state s, and thereafter selecting actions according to policy π. Under an optimal policy, π_{opt}, these functions are known as the *optimal state-value function* and the *optimal action-value function*, respectively.

It can be seen how reinforcement learning could apply to the problem of dynamic spectrum access. The radio environment consists of a set of exploitable frequency bands, each with a different characterization based on, say, DC and LZC. The state, therefore, could consist of two broad pieces of information: the frequency band the cognitive radio (agent) is currently exploiting, and all of the visible information relating to the other frequency bands including current occupancy, historical DC characterization, and historical LZ characterization. The former represents the controlled component of

the state, and the latter represents the stochastic nature of the state, where the behaviour of the licensed network users in the radio environment is modelled as a random process.

Reinforcement learning is an abstract framework, whilst MDPs are quite precisely defined mathematically. Reinforcement learning does not necessarily have to be applied to environments that satisfy the Markov property in a strict sense order to work. However, decision processes that adhere to the Markov property can be analyzed in more mathematically tractable ways. In many practical cases, the environmental information which comprises the state and is used by the agent in making its decisions can be selected so that a good approximation to an MDP results. The requirement here is that the *state needs to be sufficiently informative*, encapsulating all the information that the agent needs to make its decisions without requiring the prior history [88].

The area of reinforcement learning and associated learning algorithms is fertile. In [42], *Q-learning* is advocated, on the grounds that it does not require a model of the behaviour of the environment to be known in advance (in terms of the next-state probability distributions). However, Q-learning provides several other benefits which make it an attractive reinforcement learning approach to apply here:

- It is an off-policy approach, which has an advantage of permitting highly exploratory behaviour policy during learning, whilst still estimating the optimal policy. For example, the behaviour policy could simply be to select the next action a from the set of all possible actions $\mathcal{A}$ according to a uniform distribution.

- It is a *temporal/differential* (TD) approach [88], in that it does not have to wait until a learning episode is complete in order to evaluate the return R_t and 'learn' from it. Rather, policy-related information is learned with each epoch. This incremental approach can save learning time and computation, and is suited to problems where there is no terminal state.

A summary of some Q-learning concepts now follows.

Q-learning is based on attempting to learn the optimal action-value function by an iterative update which occurs each epoch. Initially, the action-value function, $Q(s, a)$,

is set to be 0 for all s, a. The agent is placed in an initial state, s_0, at time $t = 0$, chosen at random. Some stochastic behaviour policy is used to ensure exploration and to select the next action at each epoch. At time t, action a_t is taken in state s_t, a new state s_{t+1} results and an immediate utility value r_{t+1} is obtained. The existing action-value function is then updated as follows:

$$\begin{aligned} Q(s_t, a_t) &\leftarrow Q(s_t, a_t) + \alpha \left(r_{t+1} + \gamma \left(\max_{a \in \mathcal{A}_{s_{t+1}}} Q(s_{t+1}, a) \right) - Q(s_t, a_t) \right) \\ &= (1 - \alpha) Q(s_t, a_t) + \alpha \left(r_{t+1} + \gamma \left(\max_{a \in \mathcal{A}_{s_{t+1}}} Q(s_{t+1}, a) \right) \right), \; 0 < \alpha \leq 1. \quad (5.25) \end{aligned}$$

In words, at time t, the update to the action-value function, for a given state-action combination, depends on the current maximum value of the function when given the observed state at the next epoch, s_{t+1} (i.e. considered across all actions available in state s_{t+1}). The estimated return is then based partially on what is basically a one-stage look-ahead at the maximal return that results from taking the selected action.

In (5.25), γ is the discount factor, and α is the *learning rate*. The learning rate provides a relative weighting between the existing action value (representing past information) and new information in the update. It is evident from (5.25) that $\alpha \neq 0$, otherwise the new action value would always equal the current one and no learning would take place. If $\alpha = 1$, the current action value has no bearing on the new value.

In the context of cognitive radio, the immediate utility r_{t+1} may, for example, represent the *benefits* of transmitting at some data rate between epochs, net of any *penalties* associated with interfering with a licensed network user and resources expended in switching channels.

In terms of a software implementation, the action-value function is easily implemented as a two-dimensional table or array, indexed by an integer state identifier along one dimension, and an integer action identifier along the other. The term *Q-table* will be used to refer to such a tabular arrangement of the action-value function. Furthermore, individual values in the Q-table will be referred to as *Q-values*.

Once learning is deemed completed, the *estimate* of the optimal policy, $\hat{\pi}_{opt}$, is

determined by the action that gives the maximum Q-value given each possible state:

$$\hat{\pi}_{opt}(s,a) = \begin{cases} 1 & \text{when } a = \arg\max_{a' \in \mathcal{A}_s} Q(s,a') \\ 0 & \text{otherwise.} \end{cases} \tag{5.26}$$

This prescribes the deterministic, post-learning policy that the agent should apply, in order to maximize its return when it interacts with its environment.

The iterative updates in (5.25) can be carried out over multiple learning episodes, or where no terminal state exists, over many epochs within a non-terminal learning episode. In either case, a question that needs to be answered is: when can learning and exploration be deemed complete? Conceptually, in order for a learned outcome to be valid, it must *converge* towards some optimal value, so in this case the estimated policy must demonstrate some sign of such convergence having taken place. In addition, it is highly desirable that sufficient exploratory coverage of the state-action space should have occurred. Therefore, one possible approach is to declare learning complete when *both* of the following conditions are true:

- The estimate of the optimal policy given in (5.26) has not changed for some integer number of consecutive epochs, T_{conv}; and
- Each entry in the Q-table has been updated at least a minimum number of times, N_U.

5.3.2 Framing Channel Selection as a Q-Learning Problem

Now that reinforcement learning and Q-learning have been described, a decomposition of the channel-selection problem into a Q-learning framework is made.

Modelling States and Actions

Implicit in the discussion throughout section 5.3.1 is an assumption that the number of both the states and the available actions are finite. Where this is true, the state-value and action-value functions are both discrete, and the size of the Q-table is finite. If,

as indicated in section 5.3.1, the state consists of the occupancy status, DC estimate and LZC estimate of all the channels in the radio environment, it becomes apparent that, even for a modest number of channels, a state-space 'explosion' will result. For example, as occupancy is a binary variable, for a radio environment with C channels, the total number of states is 2^C, and this is before the number of possible values of DC and LZC for each channel are taken into account! The number of actions (channel selected, with not transmitting as also one of the possible actions) increases linearly with the number of channels considered in the radio environment, but is nonetheless potentially large.

In reinforcement learning, there are problems where the nature of the states and actions are naturally *continuous* rather than discrete. For problems such as these (where the number of states and/or actions would be infinite), or ones where the number of states and/or actions is so large that creating a complete Q-table for learning is not practical, one approach is to conduct learning in relation to a manageable subset of states and actions, but the state-value function and action-value function is represented by a relatively small vector of parameters of a function, and these parameters are adjusted each time epoch, based on the estimated values during learning. These parameters could be, for example, coefficients of a polynomial, or a collection of weight values in a neural network [88]. This approach allows the learning outcome to be *generalized* to states and actions not explicitly evaluated during training, because the parameters are deemed to provide a good approximation for the entire state-value function and action-value function. This then becomes an instance of *supervised learning* [88], but like any learning problem involving generalization, the ability to do so may be compromised by *overfitting*.

Therefore, rather than catering for these extra potential complications, the approach taken here is to *treat the channel-selection problem as one with a finite number of states, but seek out strategies for keeping the number of states tractable.* Furthermore, the value of simpler implementations that have a low computational and memory burden in software-defined radios has already been enunciated.

Quantizing Duty Cycle and Complexity Values

The values of both DC and LZC, for occupancy sequences, are *quantized* based on the length N of the sequence, and the number of quantization levels varies linearly with the *power* of N. This observation makes apparent the potential for state-space and action-space explosion as N increases.

To resolve this, the following classification scheme is applied to both DC and LZC, which effectively provides a three-stage quantization:

- **Low**: Less than 0.3
- **Medium**: Between 0.3 and 0.7 inclusive
- **High**: Greater than 0.7

Whilst the boundaries of these quantization levels are fairly arbitrary, their selection was guided by the range and distribution of values shown in Figure 5.1. There are now only nine possible ways that a channel can be characterized based on both DC and LZC.

Neglecting Channels with High Duty Cycles

Channels with high duty cycles, by definition, provide the least amount of opportunity for utilization by a cognitive radio, as they are occupied by a licensed network user most of the time.

By assuming that all channels which have high *DC (as defined above) present so little opportunity for utilization that they can be ignored*, the number of actions available to the cognitive radio is further reduced.

Decomposing into Multiple Learning Problems

Even with the measures taken so far, it is conceivable that the size of the Q-tables required could still be prohibitively large.

For example, consider a radio environment where there are C channels that the cognitive radio can utilize. Suppose that each channel's 'state', s_{chan}, may be defined by the following tuple:

$$\begin{aligned} s_{chan} &\triangleq (\mathrm{DC}_{cat}, \mathrm{LZC}_{cat}, O)\,, \\ \mathrm{DC}_{cat} &\in \{\text{low, medium, high}\}\,, \\ \mathrm{LZC}_{cat} &\in \{\text{low, medium, high}\}\,, \\ O &\in \{0, 1\}\,, \end{aligned} \tag{5.27}$$

where DC_{cat} is the category of the DC value, LZC_{cat} is the category of the LZC value and O is the channel's occupancy at the most recent epoch.

Furthermore, let the 'state' of the cognitive radio, s_{CR}, be defined by the channel that it is currently utilizing:

$$s_{CR} \in \{0, 1, 2, 3, \ldots, C\}\,, \tag{5.28}$$

where $s_{CR} = 0$ means that the cognitive radio has entered a quiescent state and is *not transmitting on any channel*, whilst $s_{CR} = c$ means that the cognitive radio is currently utilizing channel c, for $1 \leq c \leq C$.

Now, if the total state space is the amalgam of each channel's state and the cognitive radio's state, then the size of the state space is:

$$|\mathcal{S}| = (3 \times 3 \times 2)^C \times (C + 1) = 18^C\,(C + 1)\,, \tag{5.29}$$

which is unreasonably large for any realistic value of C.

To deal with this problem, the conceptual approach is to break the learning problem down to multiple smaller learning problems, where *the state space of each learning problem is manageable*, and then amalgamate the learning outcome of each of these at the conclusion.

If each channel is characterized by its DC category (DC_{cat}) and LZC category (LZC_{cat}), and by ignoring high DC channels, then at each epoch, the cognitive radio has the option of utilizing a channel belonging to one of the following six *archetypes*:

1. Low DC, Low LZC
2. Medium DC, Low LZC
3. Low DC, Medium LZC
4. Medium DC, Medium LZC
5. Low DC, High LZC
6. Medium DC, High LZC

In learning a channel selection policy, the ultimate objective is to determine the relative merit of selecting a channel belonging to a particular archetype (assuming one is present), compared to one of another archetype. Therefore, it is possible to set up more constrained learning problems, where the radio environment in each is constrained to containing channels belonging to two distinct archetypes only. *The learning outcome is a Q-table which indicates which of the two archetypes provides more long-term utility, where the cognitive radio is given a choice between the two.*

One instance of the above constrained learning problem is set up for each possible unordered pair of channel archetypes, of which there are $\binom{6}{2} = 15$.

For each of these constrained learning problems, *two channel instances of each of the two archetypes are present in the radio environment, for a total of four channels. This is to give the cognitive radio the option of switching to another channel of the same archetype as the one that it is currently utilizing, as opposed to remaining with its current channel*, as it is conceivable that the average utility will vary between these two actions, all else being equal.

The 'state' of each of these four channels is defined by its occupancy at a given epoch. Suppose the four channels in the radio environment are labelled as follows: $c_{A,1}, c_{A,2}, c_{B,1}, c_{B,2}$, where $c_{A,k}$ denotes the kth channel belonging to archetype A and

$c_{B,k}$ denotes the kth channel belonging to archetype B. Therefore, the state of the radio environment is encapsulated within the following tuple:

$$\begin{aligned} \mathcal{S} &\triangleq \left(O_{c_{A,1}}, O_{c_{A,2}}, O_{c_{B,1}}, O_{c_{B,2}}, s_{CR}\right), \\ O_{c_{A,1}} &\in \{0,1\}, \\ O_{c_{A,2}} &\in \{0,1\}, \\ O_{c_{B,1}} &\in \{0,1\}, \\ O_{c_{B,2}} &\in \{0,1\}, \\ s_{CR} &\in \{\text{idle}, c_{A,1}, c_{A,2}, c_{B,1}, c_{B,2}\}, \end{aligned} \tag{5.30}$$

where $O_{c_{A,k}}$ is the occupancy of channel $c_{A,k}$ in the current epoch, $O_{c_{B,k}}$ is the occupancy of channel $c_{B,k}$ in the current epoch, and s_{CR} denotes the channel that the cognitive radio is currently utilizing, with 'idle' indicating that it is not transmitting on any channel. Therefore, the number of states in each constrained learning problem is $|\mathcal{S}| = 2 \times 2 \times 2 \times 2 \times 5 = 80$.

In addition, the actions available to the cognitive radio correspond directly to each of the possible channels that it can utilize (with 'idle' also being an option), so the action space is given by:

$$\mathcal{A} \triangleq \{\text{idle}, c_{A,1}, c_{A,2}, c_{B,1}, c_{B,2}\}, \tag{5.31}$$

and there are five possible actions. Therefore, the Q-table can be represented by a 80×5 matrix.

Utility Function

As described previously, reinforcement learning in general relies on a utility value or reward being received by an agent interacting with its environment at each epoch, and this value depends on the previous action taken and the response of the environment to that action. Here, a suitable formulation for utility, as a function of a cognitive radio's

actions and radio environment states, is presented.

Conceptually, the actions of a cognitive radio performing dynamic spectrum access seek to utilize, as efficiently as possible, a vacant channel. In choosing to utilize a channel at a particular epoch, the device may receive a maximum *benefit* associated with this communication opportunity, B_c.

However, this benefit may also be diminished by the following *costs*:

- *Interfering with* a licensed network user on a channel that the cognitive radio is utilizing for communications, for some fraction of the time period between epochs. Let this be denoted $C_{i,out}$.

- Degradation of its own communications due to in-bound interference exerted from a licensed network user that tries to transmit on a channel that the cognitive radio is utilizing, which lasts for some fraction of the time period between epochs. Let this be denoted $C_{i,in}$.

- The cost of switching channels, where the last action required this to occur. Let this be denoted C_s.

The formulation of each of these terms will now be individually discussed. Conceptually, the basic unit of 'benefit' is the amount of data transmitted between epochs by the cognitive, whereas the unit of 'cost' is the amount of data that is *not* transmitted (or forgone) between epochs. These principles guide the following discussion.

B_c refers to the utility or benefit associated with transmitting on a channel for the time period between epochs, in the optimal scenario of no degradation due to interference from a licensed network user operating on that same channel. This benefit is sacrificed if the cognitive radio decides to go into a quiescent state for this time period (i.e. it does not transmit at all). Therefore, B_c at epoch t can be defined as follows:

$$B_c = k_{tx} R T_d, \tag{5.32}$$

where T_d is the average time period between epochs (where the cognitive radio can make a new decision), R is the achievable data rate of the cognitive radio, and k_{tx} is a binary value indicating whether or not the cognitive radio was transmitting or in a quiescent state when the last decision was made:

$$k_{tx} = \begin{cases} 0 & \text{if the cognitive radio is not transmitting on any channel i.e. } a_{t-1} = \text{idle} \\ 1 & \text{if the cognitive radio is transmitting on a channel.} \end{cases} \tag{5.33}$$

$C_{i,out}$ relates to the cost or *penalty* associated with exerting interference upon a primary user. In order to compare this with B_c, a scaling factor k_i is defined which reflects the relative 'importance' of one unit of data transmitted by the the licensed network user, compared to one unit of data transmitted by the cognitive radio. Furthermore, the following assumptions are made:

- When both the cognitive radio and the licensed network user 'collide' (i.e. both attempt to transmit on the same channel), there is sufficient *mutual interference* that the effective data throughput by *both* devices approaches zero.

- If a collision starts to occur at some point in between epochs, it is assumed (pessimistically) that neither the cognitive radio or the licensed network user has achieved any effective data throughput for the *entire* period, T_d. (Due to the discretized nature of the decision-making, it is not possible to identify the exact point in time when the collision began.)

- Therefore, at a given epoch t, if a channel is found to be occupied by a licensed network user, it is assumed that it has been transmitting for the entire time duration between epochs $t-1$ and t.

Now $C_{i,out}$ may be defined as:

$$C_{i,out} = k_{tx} o_t k_r k_i R T_d, \tag{5.34}$$

where k_r is the ratio between the data rate of the licensed network user when it transmits on a channel and that of the cognitive radio, and o_t is the occupancy of the channel by the licensed network user at epoch t (1 if occupied, 0 if not occupied). Note that, when the cognitive radio has chosen to go into a quiescent state, $k_{tx} = 0$, so $C_{i,out} = 0$, since the cognitive radio cannot exert any interference if it is not transmitting.

$C_{i,in}$ quantifies the loss of a cognitive radio's utility due to a licensed network user transmitting on the same channel and interfering with it. Due to the assumption of mutual interference, the resulting formulation is similar to $C_{i,out}$, with the only difference being the scaling factor k_i:

$$C_{i,in} = k_{tx} o_t R T_d. \tag{5.35}$$

Again, when the cognitive radio is not transmitting, $k_{tx} = 0$, so $C_{i,in} = 0$, since the cognitive radio cannot be interfered with if it is not transmitting.

C_s represents the *opportunity cost* of switching to another channel by the cognitive radio. In order to do so, it must *abandon its current channel* and then *re-establish a network link* on another channel [25] – actions which require it to sacrifice the opportunity to transmit data for part of the time between epochs, as they each require a certain amount of time to accomplish. Let the time for the cognitive radio to abandon its current channel be T_a, and the time taken to re-establish a network connection on another channel be T_e. Then:

$$C_s = k_{tx} k_s R \left(T_a + T_e \right), \tag{5.36}$$

where k_s is a binary value indicating whether or not the cognitive radio effected a channel switch in its previous action:

$$k_s = \begin{cases} 0 & \text{if the cognitive radio did not switch channels in the previous epoch} \\ 1 & \text{if the cognitive radio switched channels in the previous epoch.} \end{cases} \tag{5.37}$$

The net utility r_t takes each of these into account, whilst dividing out the cognitive radio's data rate, R, as it is common to all terms and is non-zero, and is re-evaluated at every epoch t:

$$\begin{aligned} r_t &= \frac{B_c - C_{i,out} - C_{i,in} - C_s}{R} \\ &= k_{tx}T_d - k_{tx}o_t k_r k_i T_d - k_{tx}o_t T_d - k_{tx}k_s \left(T_a + T_e\right) \\ &= k_{tx}\left(T_d\left(1 - o_t\left(k_r k_i + 1\right)\right) - k_s\left(T_a + T_e\right)\right). \end{aligned} \tag{5.38}$$

5.3.3 Summary of Q-Learning Approach for Channel Selection by a Cognitive Radio

The approach used to learn a channel selection policy using Q-learning, for use by a cognitive radio, incorporates all of the ideas from section 5.3.2. The main points are summarized below. The approach requires two stages:

- Q-learning execution
- Interpretation and policy determination

FIGURE 5.3 provides an illustration of the steps described in this section.

Q-learning Execution

Rather than formulating one learning problem with an intractable number of states and actions, $\binom{6}{2} = 15$ individual learning problems are employed instead, with each involving channels belonging to exactly *two* of the following archetypes:

- Low DC, Low LZC
- Medium DC, Low LZC
- Low DC, Medium LZC
- Medium DC, Medium LZC

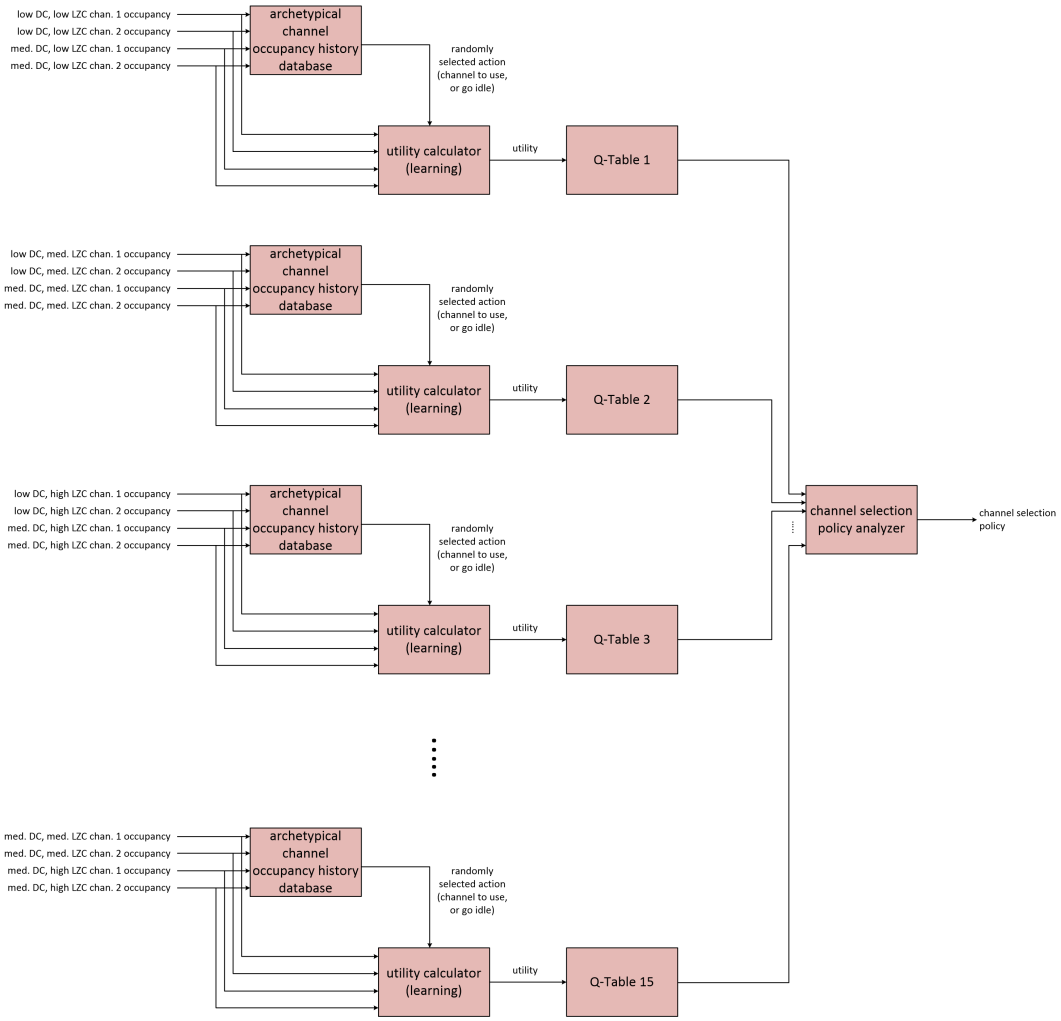

FIGURE 5.3: Decomposition of the problem of learning a channel selection policy into 15 separate Q-learning problems. Each Q-learning problem involves two channels of two different channel archetypes (section 5.3.3). The 15 Q-tables which result are collated and analyzed to determine a channel selection policy (section 5.3.3).

- Low DC, High LZC
- Medium DC, High LZC

Therefore, *the process in this section must be completed for each possible pair of archetypes.* There are two channels present in the radio environment for each archetype.

The state space of the learning problem is as shown in (5.30), and the action space is as shown in (5.31), resulting in an 80×5 Q-table.

The Q-table is initialized so that each entry, $Q(s, a)$, is equal to 0. The *initial state* of the learning problem is determined by the initial occupancies on each of the four

channels, as well as the value of s_{CR}, which is *chosen randomly according to a uniform distribution* from each of the possible values (see (5.30)).

The Q-learning algorithm is implemented in a loop, with each iteration in software corresponding with one epoch. Each iteration involves doing the following at epoch t:

1. An action a_t is selected randomly from the set $\mathcal{A}$, as defined in (5.31), according to a *uniform distribution*.

2. The traffic occupancies of each of the four channels is updated, through advancing time by one epoch.

3. The immediate utility available to the cognitive radio, r_{t+1}, is evaluated according to (5.38).

4. $Q(s_t, a_t)$, the Q-value in the Q-table corresponding to the previous state and action, is updated according to (5.25).

As there is no terminal state, these steps are repeated in a loop, until the Q-table is deemed to have converged, according to the criteria explained at the end of section 5.3.1. The *final Q-table*, as it stands at the final epoch, is saved for the next stage. These steps are summarized in Algorithm 5.1.

One problem that has not been addressed is this: how does one 'synthesize' an occupancy sequence that fulfills the DC and LZC criteria implied by each of the archetypes? It is a simple matter to take any occupancy sequence and evaluate its DC and LZC. It is also simple to generate a binary occupancy sequence with a certain value of DC, as long as one doesn't care what its LZC is. However, at this point, it is not apparent how to generate an occupancy sequence to meet a target LZC specification.

To solve this problem, the approach taken was to *search* a corpus of occupancy sequences obtained from real-world measurements for sequences of a desired length N whose DC and LZC values meet the needs of the channel archetype. The selected corpus was that provided by RWTH Aachen University as part of its spectrum occupancy measurement campaign [87], conducted between December 2006 and July 2007 at three locations, and spanning the RF spectrum from 20 - 6000 MHz:

Algorithm 5.1 Q-learning algorithm comparing two channel archetypes.

Require: Two channels with DC, LZC fulfilling archetype A $(c_{A,1}, c_{A,2})$
Require: Two channels with DC, LZC fulfilling archetype B $(c_{B,1}, c_{B,2})$
Require: N_U defined (see section 5.3.1)
Require: T_{conv} defined (see section 5.3.1)
  converged $\leftarrow$ false
  samecount $\leftarrow 0$ {*Counts no. epochs policy is unchanged*}
  $\mathcal{A} \leftarrow \{idle, c_{A,1}, c_{A,2}, c_{B,1}, c_{B,2}\}$ {*Set of cognitive radio actions*}
  $\mathbf{s_{CR}} \leftarrow$ random value from $\mathcal{A}$
  Evaluate occupancies of each channel: $\mathbf{O_{c_{A,1}}}, \mathbf{O_{c_{A,2}}}, \mathbf{O_{c_{B,1}}}, \mathbf{O_{c_{B,2}}}$
  $\mathbf{s_t} \leftarrow \left(\mathbf{O_{c_{A,1}}}, \mathbf{O_{c_{A,2}}}, \mathbf{O_{c_{B,1}}}, \mathbf{O_{c_{B,1}}}, \mathbf{s_{CR}}\right)$
  $\mathbf{Q} \leftarrow 80 \times 5$ matrix initialized to 0 {*Q-table; indexed by* s, a}
  Qcount $\leftarrow 80 \times 5$ matrix initialized to 0 {*Records no. times each corresponding Q-table entry is updated; indexed by* s, a}
  policyopt $\leftarrow 80 \times 5$ matrix initialized to 0 {*Table of policies for each state; indexed by* s, a}
  while converged = false **do**
    $\mathbf{a_{old}} \leftarrow \mathbf{a_t}$
    $\mathbf{a_t} \leftarrow$ random selection from $\mathcal{A}$ {*Randomly selected action*}
    if $\mathbf{a_{old}} \neq \mathbf{a_t}$ **then**
      $\mathbf{k_s} \leftarrow 1$ {*Channel switch took place*}
    else
      $\mathbf{k_s} \leftarrow 0$ {*Channel switch did not take place*}
    end if
    $\mathbf{s_{CR}} \leftarrow \mathbf{a_t}$
    Advance time epoch by 1 step on channels $(c_{A,1}, c_{A,2}, c_{B,1}, c_{B,2})$ {*Using archetypical channel occupancy history database*}
    Evaluate occupancies of each channel: $\mathbf{O_{c_{A,1}}}, \mathbf{O_{c_{A,2}}}, \mathbf{O_{c_{B,1}}}, \mathbf{O_{c_{B,2}}}$
    $\mathbf{s_{t+1}} \leftarrow \left(\mathbf{O_{c_{A,1}}}, \mathbf{O_{c_{A,2}}}, \mathbf{O_{c_{B,1}}}, \mathbf{O_{c_{B,1}}}, \mathbf{s_{CR}}\right)$ {*New state*}
    $\mathbf{r_{t+1}} \leftarrow$ value given by (5.38) {*Utility calculator, for current epoch*}
    $\mathbf{Q}\left(\mathbf{s_t}, \mathbf{a_t}\right) \leftarrow$ value given by (5.25) {*Q-table update (learning)*}
    $\mathbf{Qcount}\left(\mathbf{s_t}, \mathbf{a_t}\right) \leftarrow \mathbf{Qcount}\left(\mathbf{s_t}, \mathbf{a_t}\right) + 1$
    $\mathbf{s_t} \leftarrow \mathbf{s_{t+1}}$
    policyoptold $\leftarrow$ **policyopt**
    policyopt $\leftarrow$ policy given by (5.26)
    if policyoptold = **policyopt then**
      samecount $\leftarrow$ **samecount** $+ 1$
      if samecount $\geq T_{conv}$ **then**
        if Qcount $(s, a) \geq N_U \; \forall s, a$ **then**
          converged = true
        end if
      end if
    else
      samecount $\leftarrow 0$ {*Policy has changed, restart the 'policy unchanged' counter*}
    end if
  end while

- Aachen, Germany (indoors, modern office building)
- Aachen, Germany (third-floor balcony, in a residential area)
- Maastrict, Netherlands (rooftop, school in a residential area)

MATLAB scripts were written which extracted the occupancy sequences of length N from the raw data, and categorized them by DC and LZC as follows:

- **Duty cycle**: 0.0 - 0.1, 0.1 - 0.2, 0.2 - 0.3, 0.3 - 0.4, 0.4 - 0.5, 0.5 - 0.6, 0.6 - 0.7, 0.7 - 0.8, 0.8 - 0.9, 0.9 - 1.0
- **LZ complexity**: 0.0 - 0.1, 0.1 - 0.2, 0.2 - 0.3, 0.3 - 0.4, 0.4 - 0.5, 0.5 - 0.6, 0.6 - 0.7, 0.7 - 0.8, 0.8 - 0.9, 0.9 - 1.0, greater than 1.0

For each category, *index files* were generated which allowed a compliant sequence to be extracted quickly from the corpus without having to spend time searching for it. These files contained the following indexing information:

- Raw occupancy measurement file (as obtained directly from [87])
- Sub-channel ID (corresponding to a particular 200 kHz frequency band within the occupancy measurement file)
- Time offset (corresponding to a particular time-stamp in the occupancy measurement file, from which N occupancy values can then be extracted)
- Sequence length (N)

Therefore, once this indexing stage was completed, to generate occupancy values for a channel belonging to a particular archetype, it was a simple matter of reading the index files with the requisite DC and LZC categories, selecting a random index entry, and extracting the corresponding occupancy sequence from the raw measurements.

Interpretation and Policy Determination

Once Q-learning has been completed for each pair of channel archetypes, the learning outcomes from each are then interpreted to determine a channel selection policy.

In this context, the channel selection policy is a list of possible actions for the cognitive radio to take, ranked in decreasing order of preference, depending on the archetype of the channel that it is currently occupying, and whether or not it is currently occupied by a licensed network user at the epoch when the decision has to be made.[3]

From the point of view of the cognitive radio's individual 'state', from which an action must be selected, the state space is:

1. Occupying a *low DC, low LZC* channel, and *occupied* by a licensed network user
2. Occupying a *low DC, low LZC* channel, and *not occupied* by a licensed network user
3. Occupying a *low DC, medium LZC* channel, and *occupied* by a licensed network user
4. Occupying a *low DC, medium LZC* channel, and *not occupied* by a licensed network user
5. Occupying a *low DC, high LZC* channel, and *occupied* by a licensed network user
6. Occupying a *low DC, high LZC* channel, and *not occupied* by a licensed network user
7. Occupying a *medium DC, low LZC* channel, and *occupied* by a licensed network user
8. Occupying a *medium DC, low LZC* channel, and *not occupied* by a licensed network user

[3]This is related to but distinct from the technical definition of an optimal policy in Q-learning, as shown in (5.26). The optimal policy corresponds with the top-ranked action in the list described here; lower ranked actions are employed when the optimal action is not possible.

9. Occupying a *medium DC, medium LZC* channel, and *occupied* by a licensed network user
10. Occupying a *medium DC, medium LZC* channel, and *not occupied* by a licensed network user
11. Occupying a *medium DC, high LZC* channel, and *occupied* by a licensed network user
12. Occupying a *medium DC, high LZC* channel, and *not occupied* by a licensed network user
13. Not transmitting (quiescent or idle)

Note that these states are *subsets* of the states used in the Q-learning problems from the previous stage, since these states only consider the DC and LZC characterization of at most one channel (the one the cognitive radio is utilizing). Hence, this will be referred to as the list of *reduced states*.

When the cognitive radio is utilizing a channel belonging to archetype A, it has the following *channel selection actions* available to it:

- Remain on the channel it is currently utilizing
- Switch to another channel of the same archetype A that is occupied by its licensed network user
- Switch to another channel of the same archetype A that is not occupied by its licensed network user
- Switch to another channel of archetype B that is occupied by its licensed network user, where $B \neq A$
- Switch to another channel of archetype B that is not occupied by its licensed network user, where $B \neq A$
- Stop transmitting (and become quiescent)

On the other hand, when it is not transmitting (is quiescent), the available channel selection actions are:

- Remain in a quiescent state
- Switch to a channel of some archetype A that is not occupied by its licensed network user
- Switch to a channel of some archetype A that is occupied by its licensed network user

It was stated in (5.26) that Q-values indicate the expected discounted return, defined in (5.26), from following a particular action selection policy (in the case of Q-learning used here, randomly selected from a uniform distribution). Given a certain state, the highest Q-value suggests the optimal action to take.

Therefore, the ordered list of actions, for each possible reduced state above, is determined as follows:

1. For *each* of the final Q-tables obtained from the previous stage, locate all entries whose Q-learning states s 'match' the reduced state under consideration. e.g. if considering reduced state 3 above, referring to (5.30), a state value of $s = (0, 1, 1, 0, c_{A,2})$ from the Q-table is one that matches, where A is the archetype 'low DC, medium LZC'.

2. Select an action whose utility is to be considered, and *eliminate* from the previous set of Q-table entries all those which *do not* match this action. e.g. if considering the action '*switch to a channel of archetype medium DC, medium LZC that is not occupied by its licensed network user*', then referring to (5.31) and continuing the previous example, an action of $a = c_{B,2}$ from the Q-table matches, where B is the archetype 'medium DC, medium LZC'. On the other hand, an action of $a = c_{A,1}$ does not match (since this involves switching to a channel of archetype 'low DC, medium LZC'), nor does $a = c_{B,1}$ (since this involves switching to a channel of archetype 'medium DC, medium LZC' that *is* occupied by its licensed network user).

		Case 1	Case 2
Q-learning Parameters	α	0.2	0.2
	γ	0.9	0.9
Q-learning Convergence Parameters	N_U	100	100
	T_{conv}	100	100
Utility Function Parameters	T_d	5	5
	k_r	1	2
	k_i	10	10
	T_a	0.5	0.5
	T_e	0.5	0.5

TABLE 5.1: Convergence and utility function parameters for Q-learning simulation to determine a channel selection policy.

3. For the remaining entries (which haven't been eliminated), find the average Q-value.

4. Repeat steps 2 - 3 for all possible channel selection actions.

5. Rank all of the channel selection actions by decreasing average Q-value as obtained from step 3. This forms the channel selection policy given a particular reduced state.

These steps are then repeated for every reduced state, to compile the complete channel selection policy.

It is noteworthy that the final channel selection policy will be sensitive to the utility function, r_t, used – a different function may yield a different channel selection policy.

5.3.4 Channel Selection Policy Learning Results

The approach described in section 5.3.3 was implemented in MATLAB, for two cases which differ slightly in the formulation of the utility function, r_t. This is described in TABLE 5.1.

The only difference between the two cases is that, in 'Case 2', the licensed network user on a channel transmits with a higher data rate (as indicated by k_r) compared to the cognitive radio when utilizing the same channel. With a high value of k_i, the

relative 'importance' of the licensed network user's traffic is emphasized in the utility function, so the resulting policy should tend to favour actions which avoid the costs of interfering with the licensed network user compared to the benefit gained from the cognitive radio transmitting. The values of T_a (channel abandonment time) and T_e (network re-establishment time) were taken from maximum acceptable bounds (500 ms) used in [25] when testing next-generation (XG) radio implementations for DARPA. $T_d = 5$ implies that the cognitive radio makes a decision about a (possible) channel switch every 5 seconds.

Table 5.2 and Table 5.3 show the resulting policies for 'Case 1' and 'Case 2', respectively. Each column indicates an *ordered* list of actions to take, in decreasing preference, when the cognitive radio is in a particular state (as indicated by the column's heading). The need for a preferential list like this is because some actions may not always be possible. For example, the highest-preference action may be to switch to a channel that has 'low DC, low LZC' that is also vacant, but at the epoch under consideration there may not be a channel which meets this criterion, so the next preferred action should be considered instead.

Note that, at first glance, the ordered lists may seem to terminate prematurely, with some actions denoted with '–'. It was mentioned that some actions may not always be possible, but *there are two actions that are guaranteed to be possible*:

- Where the cognitive radio is utilizing a channel, to continue to do so (i.e. 'stay' in that channel)
- Go into an idle (quiescent) state.

Therefore, *once the ordered list reaches an action that corresponds to either of these, the list can be considered complete*, and remaining actions do not need to be considered.

In general terms, the difference between the resultant policy for 'Case 1' and 'Case 2' is that Case 2 favours going into an idle state over switching to a channel with 'low DC, medium LZC'. This is because Case 2 involves the licensed network user transmitting at a higher data rate when utilizing the channel compared to the cognitive radio ($k_r = 2$), and so the cost of interfering with the licensed network user ($C_{i,out}$) is relatively high

		Cognitive Radio States												
		low DC, low LZC		low DC, med. LZC		low DC, high LZC		med. DC, low LZC		med. DC, med. LZC		med. DC, high LZC		idle
		vac.	occ.	vac.	occ.	vac.	occ.	vac.	occ.	vac.	occ.	vac.	occ.	
Cognitive Radio Actions	low DC, low LZC, vac. (stay)	1	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
	low DC, low LZC, vac. (switch)	–	1	1	1	1	1	1	1	1	1	1	1	1
	low DC, low LZC, occ. (stay)	✖	–	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
	low DC, low LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	low DC, med. LZC, vac. (stay)	✖	✖	3	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
	low DC, med. LZC, vac. (switch)	–	3	–	3	3	3	–	3	3	2	3	3	3
	low DC, med. LZC, occ. (stay)	✖	✖	✖	–	✖	✖	✖	✖	✖	✖	✖	✖	✖
	low DC, med. LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	low DC, high LZC, vac. (stay)	✖	✖	✖	✖	–	✖	✖	✖	✖	✖	✖	✖	✖
	low DC, high LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	low DC, high LZC, occ. (stay)	✖	✖	✖	✖	✖	–	✖	✖	✖	✖	✖	✖	✖
	low DC, high LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	med. DC, low LZC, vac. (stay)	✖	✖	✖	✖	✖	✖	3	✖	✖	✖	✖	✖	✖
	med. DC, low LZC, vac. (switch)	–	2	2	2	2	2	2	2	2	3	2	2	2
	med. DC, low LZC, occ. (stay)	✖	✖	✖	✖	✖	✖	✖	–	✖	✖	✖	✖	✖
	med. DC, low LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	med. DC, med. LZC, vac. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	–	✖	✖	✖	✖
	med. DC, med. LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	med. DC, med. LZC, occ. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	✖	–	✖	✖	✖
	med. DC, med. LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	med. DC, high LZC, vac. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	–	✖	✖
	med. DC, high LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	med. DC, high LZC, occ. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	–	✖
	med. DC, high LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
	go idle	–	4	–	4	4	4	–	4	4	4	4	4	4

Table 5.2: Channel selection policy, Case 1. The terms 'vacant' (vac.) and 'occupied' (occ.) refer to the occupancy status of the licensed network user on that channel. ✖ denotes an invalid action that is not applicable. – denotes an action that does not need to be considered.

Cognitive Radio Actions	Cognitive Radio States: low DC, low LZC vac.	low DC, low LZC occ.	low DC, med. LZC vac.	low DC, med. LZC occ.	low DC, high LZC vac.	low DC, high LZC occ.	med. DC, low LZC vac.	med. DC, low LZC occ.	med. DC, med. LZC vac.	med. DC, med. LZC occ.	med. DC, high LZC vac.	med. DC, high LZC occ.	idle
low DC, low LZC, vac. (stay)	1	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
low DC, low LZC, vac. (switch)	–	1	1	1	1	1	1	1	1	1	1	1	1
low DC, low LZC, occ. (stay)	✖	–	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
low DC, low LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
low DC, med. LZC, vac. (stay)	✖	✖	–	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
low DC, med. LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	3	–	–	–
low DC, med. LZC, occ. (stay)	✖	✖	✖	–	✖	✖	✖	✖	✖	✖	✖	✖	✖
low DC, med. LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
low DC, high LZC, vac. (stay)	✖	✖	✖	✖	–	✖	✖	✖	✖	✖	✖	✖	✖
low DC, high LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
low DC, high LZC, occ. (stay)	✖	✖	✖	✖	✖	–	✖	✖	✖	✖	✖	✖	✖
low DC, high LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
med. DC, low LZC, vac. (stay)	✖	✖	✖	✖	✖	✖	3	✖	✖	✖	✖	✖	✖
med. DC, low LZC, vac. (switch)	–	2	2	2	2	2	2	2	2	2	2	2	2
med. DC, low LZC, occ. (stay)	✖	✖	✖	✖	✖	✖	✖	–	✖	✖	✖	✖	✖
med. DC, low LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
med. DC, med. LZC, vac. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	–	✖	✖	✖	✖
med. DC, med. LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
med. DC, med. LZC, occ. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	✖	–	✖	✖	✖
med. DC, med. LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
med. DC, high LZC, vac. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	–	✖	✖
med. DC, high LZC, vac. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
med. DC, high LZC, occ. (stay)	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	–	✖
med. DC, high LZC, occ. (switch)	–	–	–	–	–	–	–	–	–	–	–	–	–
go idle	–	3	3	3	3	3	–	3	3	4	3	3	3

Table 5.3: Channel selection policy, Case 2. The terms 'vacant' (vac.) and 'occupied' (occ.) refer to the occupancy status of the licensed network user on that channel. ✖ denotes an invalid action that is not applicable. – denotes an action that does not need to be considered.

compared to the benefit achieved from utilizing a channel (B_c). By switching to a channel with low DC, medium LZC when it's vacant, the medium LZC suggests that there is a reasonable chance that the channel may become occupied at the next epoch, which will result in ($C_{i,out}$) being incurred. Overall, it is therefore better to actually enter an idle state, and wait for either a 'low DC, low LZC' or 'medium DC, low LZC' channel to become available. This observation reiterates the notion that *the resultant policy is sensitive to how utility is measured.*

Some other notable observations, common to both 'Case 1' and 'Case 2' are:

- When the cognitive radio is currently utilizing a channel with 'medium DC, low LZC', even if the channel is vacant, it is actually slightly better to switch to another channel with the same characteristics (provided it is also vacant) and incur the cost of switching (C_s), rather than staying in its current channel! This appears counter-intuitive, but is likely driven by the above explanation for the difference between the resultant policies for 'Case 1' and 'Case 2' – namely, that the channel it is currently utilizing has a reasonable chance of becoming occupied at the next epoch, but by picking another channel with the same characteristics, there is a reasonable chance of picking another channel which has only just recently become vacant, and is likely to stay vacant for a few more epochs due to its low DC.

- Low LZC is more of a factor as a measure of how favourable a channel is, compared to low DC. e.g. it is better to select a channel with 'medium DC, low LZC', than one with 'low DC, medium LZC'. *This demonstrates the benefit of characterizing channels in ways that are more sophisticated than simply DC*, as this observation runs counter to what one might have expected if only DC was considered.

At this point, it is worth mentioning that Q-learning takes place within an environment that is presumed static – the DC and LZC characteristics of the channels seen during learning are not time varying. This may be difficult to ensure when out in the field. Furthermore, it would be difficult to guarantee the exploratory behaviour required during learning when out in the field, as this would require channel occupancies

encompassing the entire range of DC and LZC values. This suggests that the channel selection policy would result from an offline learning process. However, because the learned process is so exploratory, the learned policy has generic applicability, rather than being environment-specific. Therefore, the learned policy would be useful even in non-static radio environments where the DC and LZC characteristics of each channel vary with time, *as long as the concept of utility that was used during learning remains applicable*.

Finally, it should be noted that if a cognitive radio was to apply a channel selection policy learned this way in actual field operation, it must be able to continually evaluate the DC and LZC values of the channels that it has to pick from, so that it can categorize them according to the required archetypes. Particularly for mobile devices subjected to varying radio environments, the DC and LZC characteristics of channels may change over time. Therefore, there is a motivation to seek out a way to reduce the computational complexity of calculating the LZC of an occupancy sequence belonging to a channel, as it varies over time. The next section considers this problem.

5.4 Fast Calculation of LZ Complexity

A MATLAB implementation of a function to calculate this for a binary sequence was written and, when scrutinized under the MATLAB Profiler, it was found that most of the computational effort was expended in evaluating the sequence's eigenfunction. Despite employing various optimization strategies based on exploitation of the theorems and lemmas published in [43], the calculation of the eigenfunction was still very slow compared to calculating the DC. This is because the process involves a significant number of substring search operations on the occupancy sequence.

If a cognitive radio terminal were to operate in a static radio environment, where the underlying occupancy statistics of the licensed network users in each channel of interest remained stationary, then it could simply evaluate DC and LZC for each channel on a sufficiently long occupancy sequence observation, and it could proceed indefinitely assuming that the DC and LZC characterizations of each channel would not change

in the future. The high cost of an LZC calculation would be amortized over time, and could be largely neglected. However, this is unrealistic, particularly if considering radio environments where terminals are mobile. Different geographical areas may have different licensed network users with different traffic patterns, and different occupancy patterns may be seen in each channel as licensed network users move around over time. Even if the wireless terminals were fixed in location, it's possible that transmission characteristics may vary depending on, for example, the time of day (e.g. peak and off-peak traffic).

It is then apparent that, for a cognitive radio to apply a channel selection policy based on LZC, it must be able to continually evaluate this for the observed occupancy sequences associated with each frequency band or channel of interest. Hence, any measure that would facilitate a significant reduction in the amount of computation required to evaluate the eigenfunction of an occupancy sequence would be very useful for cognitive radio applications employing the approaches described so far.

Conceptually, one could view any channel as having an infinite-length occupancy sequence covering all time in the past and in the future. A cognitive radio terminal observes a length-N *windowed* version of this occupancy sequence. This window is a *sliding window* so that, with each time epoch, the occupancy sequence changes, but in an *incremental* way:

- The occupancy value associated with the oldest epoch is discarded
- A new occupancy value associated with the most recent epoch is appended to the remaining sequence

This is illustrated in FIGURE 5.4. Another way of visualizing the sequence is as a *first in, first out* (FIFO) queue of occupancy values, with fixed length N. Subsequently, the change to the occupancy sequence seen with each epoch will be referred to as a *single-step update*. The DC and LZC values of the occupancy sequence also change incrementally as a result of this update process. For example, if occupancy values of 0 are continually encountered and appended, then both DC $\rightarrow 0$ and LZC $\rightarrow 0$ from their initial values with each passing epoch, until the sequence consists of all 0s.

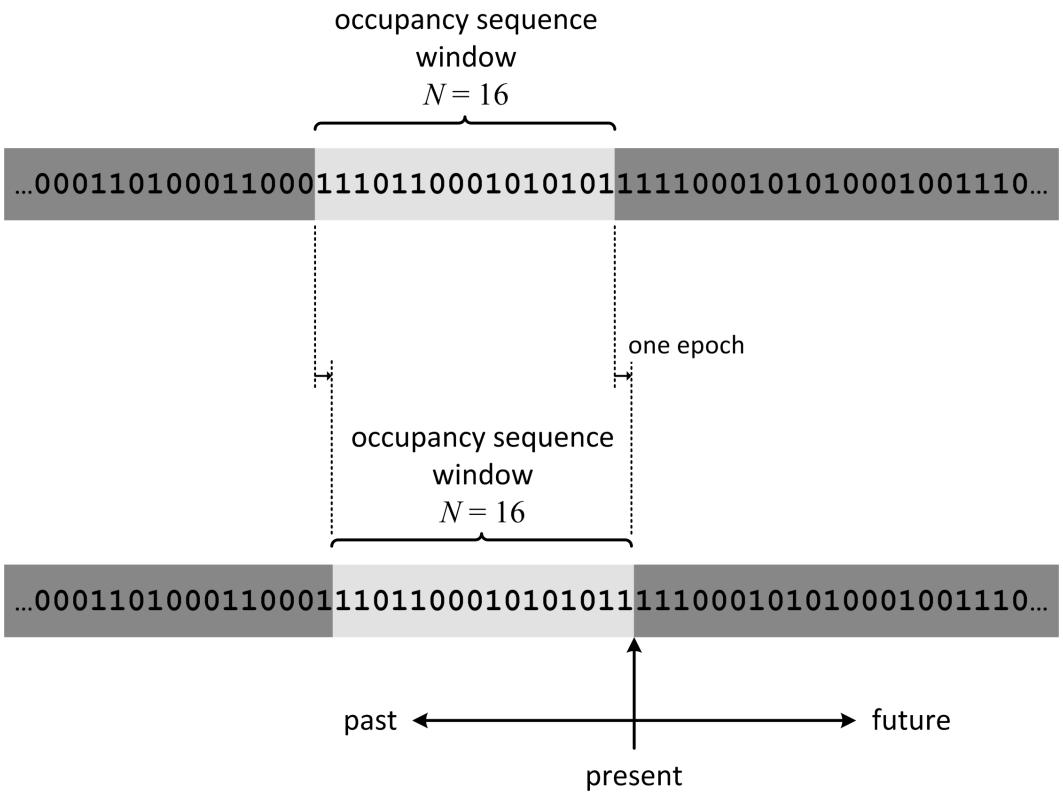

Figure 5.4: An occupancy sequence as a finite-length windowed observation of an infinite-length sequence, subject to a single-step update as the window slides to the right with each passing epoch.

It is intuitively plausible that, with a single-step update, there is only an incremental change to the vocabularies and eigenvocabularies of the sequence as the sequence index value increases, when compared to the sequence prior to the update. As it is the cardinality of the sequence's eigenvocabulary, as a function of sequence index, that determines the eigenfunction (see section 5.2.2), it also seems reasonable that most of the computation that was expended in calculating the eigenfunction of the sequence prior to the update is still valid, and only an incremental calculation is needed to obtain the eigenfunction for the present sequence. In the remainder of this section, this possibility is considered, resulting in an approach for efficiently calculating the eigenfunction of an occupancy sequence that is subject to a single-step update, *given that the eigenfunction of the sequence prior to the single-step update is known.*

Let an initial occupancy sequence, $\mathbf{o}$ be represented by a length-N sequence of

occupancy values:

$$\begin{array}{lccccccc} \mathbf{o} = & o[1] & o[2] & o[3] & \ldots & o[N-1] & o[N] \\ k = & 1 & 2 & 3 & \ldots & N-1 & N \end{array} \tag{5.39}$$

with k representing the *position* index of terms in the sequence, and $o[i] \in \{0,1\}, 1 \leq i \leq N$. Henceforth, o_k denotes the kth term of $\mathbf{o}$. Once this sequence is subjected to a single-step update, a new sequence, $\mathbf{o}^1$, results:

$$\begin{array}{lccccccc} \mathbf{o}^1 = & o[2] & o[3] & o[4] & \ldots & o[N] & o[N+1] \\ k = & 1 & 2 & 3 & \ldots & N-1 & N \end{array} \tag{5.40}$$

with $o[j] \in \{0,1\}, 2 \leq j \leq N+1$. Similarly, o_k^1 denotes the kth term of $\mathbf{o}^1$. The aim is to see if the eigenfunction of $\mathbf{o}^1$, $f_{\mathbf{o}^1}^{\lambda}[k]$, can be computed from the eigenfunction of $\mathbf{o}$, $f_{\mathbf{o}}^{\lambda}[k]$ (see (5.8)). Henceforth, the phrase 'old sequence' refers to $\mathbf{o}$, and 'new sequence' refers to $\mathbf{o}^1$.

Some additional terms relating to sequences before and after a single-step update are introduced here, to aid the following discussion. It is obvious that $\mathbf{o}^1$ is effectively a left-shifted version of $\mathbf{o}$, and that there are at least $N-1$ occupancy values that remain the same after the single-step update. Referring to (5.39) and (5.40), the term *corresponding positions* is used to describe position indices in the two sequences which represent the same occupancy value. For example, $o_2 = o_1^1 = o[2]$, so position 1 in sequence $\mathbf{o}^1$ *corresponds* with position 2 in sequence $\mathbf{o}$, position 2 in sequence $\mathbf{o}^1$ *corresponds* with position 3 in sequence $\mathbf{o}$, and in general, position k in sequence $\mathbf{o}^1$ *corresponds* with position $k+1$ in sequence $\mathbf{o}$ for $1 \leq k \leq N-1$.

The following observation can be made: *for any sequence, the value of the eigenfunction is equal to 1 up until the first position index whose sequence value is not equal to the first symbol.* Equivalently, *the value of the eigenfunction is equal to 1 up until the first position index where a second unique symbol occurs.* More formally:

$$f_{\mathbf{o}}^{\lambda}[k] = 1 \text{ for } 1 \leq k < k'$$
$$\text{where } k' \text{ is the first index such that } o_{k'} \neq o_1. \tag{5.41}$$

More visually:

$$\begin{array}{rccccccccc} \mathbf{o} = & a_1 & a_1 & a_1 & \ldots & a_1 & a_2 & a_3 & \ldots \\ k = & 1 & 2 & 3 & \ldots & k'-1 & k' & k'+1 & \ldots \\ f_{\mathbf{o}}^{\lambda}[k] = & 1 & 1 & 1 & \ldots & 1 & k' & k' & \ldots \end{array} \tag{5.42}$$

where a_n denotes the nth term from the symbol alphabet. It is easy to visualize that, in general, for any sequence with length k, where all terms are the same (a_1):

- The vocabulary of the sequence consists of sub-sequences of length 0 to k inclusive with all terms also being the same (a_1)
- The eigenvocabulary of the sequence consists of only the sequence itself (i.e. the sub-sequence with length k).

Hence the cardinality of the eigenvocabulary is equal to 1 for $1 \leq k < k'$, and this assertion is justified.

Another observation can be made: *at the first position index where the sequence value is not equal to the first sequence value, the eigenfunction is equal to the position index*. More formally:

$$f_{\mathbf{o}}^{\lambda}[k'] = k'. \tag{5.43}$$

Again, this is apparent by inspection. The introduction of a differing term at index k', a_2, adds exactly k' new sub-sequences which contain the term a_2 to the vocabulary of $\mathbf{o}_{1,k'}$ – namely, all sub-sequences which end in a_2 (being $\mathbf{o}_{k',k'}, \mathbf{o}_{k'-1,k'}, \mathbf{o}_{k'-2,k'}, \ldots, \mathbf{o}_{2,k'}, \mathbf{o}_{1,k'}$). It is obvious that these k' sub-sequences all belong in the eigenvocabulary of $\mathbf{o}_{1,k'}$, since the a_2 term is not present in any proper prefix. Also, the statement made in [43] (immediately after the proof to Lemma 3) says that for any sequence $\mathbf{o} \neq \varnothing$ with length

N, if and only if the last value is different from all its predecessor values, then the cardinality of the sequence's eigenvocabulary is given by $|E(\mathbf{o})| = N$. When applying this reasoning to sub-sequence $\mathbf{o}_{1,k'}$, the assertion being made logically follows.

Applying this reasoning to both the old and new sequences, and taking into account the relationship between corresponding positions:

$$\begin{array}{rccccccccc} \mathbf{o}^1 = & a_1 & a_1 & a_1 & \dots & a_1 & a_2 & a_3 & \dots \\ k = & 1 & 2 & 3 & \dots & k'-2 & k'-1 & k' & \dots \\ f^{\lambda}_{\mathbf{o}^1}[k] = & 1 & 1 & 1 & \dots & 1 & k'-1 & k'-1 & \dots \end{array} \tag{5.44}$$

Hence:

$$f^{\lambda}_{\mathbf{o}^1}[k'-1] = k'-1. \tag{5.45}$$

By definition, this means that the eigenvocabulary of $\mathbf{o}_{1,k'}$ has one more term than the eigenvocabulary of $\mathbf{o}_{1,k'-1}$:

$$\begin{aligned} E(\mathbf{o}_{1,k'}) &= \{\mathbf{o}_{1,k'}, \mathbf{o}_{2,k'}, \mathbf{o}_{3,k'}, \dots, \mathbf{o}_{k'-1,k'}, \mathbf{o}_{k',k'}\} \\ E\left(\mathbf{o}^1{}_{1,k'-1}\right) &= \left\{\mathbf{o}^1{}_{1,k'-1}, \mathbf{o}^1{}_{2,k'-1}, \mathbf{o}^1{}_{3,k'-1}, \dots, \mathbf{o}^1{}_{k'-2,k'-1}, \mathbf{o}^1{}_{k'-1,k'-1}\right\} \\ &= \{\mathbf{o}_{2,k'}, \mathbf{o}_{3,k'}, \mathbf{o}_{4,k'}, \dots, \mathbf{o}_{k'-1,k'}, \mathbf{o}_{k',k'}\}. \end{aligned} \tag{5.46}$$

The difference between the two is the sub-sequence $\mathbf{o}_{1,k'}$.

Referring back to (5.42), it can be seen that:

$$f^{\lambda}_{\mathbf{o}}[k'+1] = k'. \tag{5.47}$$

This can be seen as follows: regardless of the value of a_3, the sub-sequence $\mathbf{o}_{k',k'+1} = a_2a_3$ is guaranteed not to be in the vocabulary of any proper prefix of $\mathbf{o}_{1,k'+1}$, because the said vocabulary can only consist of sub-sequences where:

- Symbol a_2 does not exist; or
- Symbol a_2 is the last symbol in the sub-sequence

Therefore, sub-sequence $\mathbf{o}_{k',k'+1} = a_2 a_3$ is in the eigenvocabulary of $\mathbf{o}_{1,k+1}$. In addition, *Theorem 6* in [43] says that, for $1 \leq k \leq k'$, $\mathbf{o}_{k,k'+1}$ must also then be in the eigenvocabulary. Therefore, the eigenvocabulary has a cardinality of k', so the assertion follows. Similar reasoning leads to the conclusion that for the new sequence:

$$f^{\lambda}_{\mathbf{o}^1}[k'] = k' - 1. \tag{5.48}$$

At this point, it may be tempting to generalize that, for position indices $k \geq k' - 1$, the eigenfunction value in the new sequence will always be one less than that the eigenfunction value at the corresponding position in the old sequence. This is not plausible, since it would imply that, with each single-step update, the LZC value of the sequence would gradually approach 0.

So, when does this relationship not hold? Firstly it is useful to consider why this differential of 1 exists in the first place – it's because the sub-sequence $\mathbf{o}_{1,k'}$ has been removed from the vocabulary of ${\mathbf{o}^1}_{1,k}$, due to the removal of the leading symbol in $\mathbf{o}$ after the single-step update, for the position index values of k considered so far.

However, what happens if the proper prefix of $\mathbf{o}$, $\mathbf{o}_{1,k'}$ then *reappears* later on in sequence $\mathbf{o}^1$? Some thought shows that, for a subsequent position $k = k''$ in the new sequence, where sub-sequence $\mathbf{o}_{1,k'}$ is a *suffix* of ${\mathbf{o}^1}_{1,k''}$, this sub-sequence gets added to the eigenvocabulary of ${\mathbf{o}^1}_{1,k}$, because this marks the first occasion where the sub-sequence $\mathbf{o}_{1,k'}$ has appeared in $\mathbf{o}^1$. Therefore, it cannot be part of the vocabulary of any proper prefix of ${\mathbf{o}^1}_{1,k}$. On the other hand, at the corresponding position $k = k'' + 1$ in $\mathbf{o}$, it has already appeared (right at the start), so it is in the vocabulary of at least one proper prefix of $\mathbf{o}_{1,k''+1}$, and therefore not in its eigenvocabulary. At this position, the new sequence has *gained* a new entry to its eigenvocabulary relative to the old sequence at the corresponding position, *negating the differential of 1 that occurred earlier on.*

It is conceivable that this phenomenon could also occur with other proper prefixes of $\mathbf{o}$ besides $\mathbf{o}_{1,k'}$, with length greater than k' – they are not proper prefixes of $\mathbf{o}^1$ due to the single-step update, but reappear at a later position. However, the search for such proper prefixes can be constrained by employing Theorem 6 in [43]. Theorem

6 provides a quick way of determining the eigenvocabulary of a sequence when its eigenvalue is known – it is the list of suffixes whose starting position is less than or equal to the eigenvalue. A corollary of this is *the length of the shortest sub-sequence in the eigenvocabulary is one more than the difference between the length of the sequence and the sequence's eigenvalue.*

The goal is to search for positions where the corresponding eigenvalue needs to be *increased* (by 1), due to a proper prefix $\mathbf{o}_{1,k'}$ of $\mathbf{o}$ re-appearing as a suffix of ${\mathbf{o}^1}_{1,k''}$ and hence being included into its eigenvocabulary. Where this is true, the shortest sub-sequence in the eigenvocabulary is then equal to the difference between the length of the sequence ${\mathbf{o}^1}_{1,k''}$ and the sequence's eigenvalue. *This difference places an upper limit on the length of the proper prefix whose re-emergence later in the sequence needs to be searched for.* It is not necessary to search for proper prefixes longer than this difference, since it would imply that the eigenfunction at a matching location either does not need to be changed (rendering the search redundant), or needs to be reduced (which leads to a contradiction in Theorem 6).

The fast calculation approach can be summarized by the following steps. The position index k' is determined in the old sequence. For position indices less than the corresponding position index in the new sequence ($k < k' - 1$), the eigenfunction is 1. For other position indices in the new sequence ($k \geq k' - 1$), it is assumed initially that the eigenfunction value is one less than the eigenfunction value for the corresponding position in the old sequence. Then, for a limited set of proper prefixes of the old sequence, the position where they re-appear later in the new sequence is determined. Where this search is successful, the eigenvalue corresponding to the end position in the new sequence of the located sub-sequence is incremented by 1.

Algorithmically, the procedure for evaluating eigenfunction $f^{\lambda}_{\mathbf{o}^1}[k]$, given that the eigenfunction $f^{\lambda}_{\mathbf{o}}[k]$ is already known, can be expressed thus:

1. Locate the position index k' in $\mathbf{o}$ where the second unique symbol value occurs.

2. For position indices $1 \leq k < k' - 1$ in $\mathbf{o}^1$, set $f^{\lambda}_{\mathbf{o}^1}[k] = 1$.

3. For position indices $k' - 1 \leq k < N$ in $\mathbf{o}^1$, set $f^{\lambda}_{\mathbf{o}^1}[k] = f^{\lambda}_{\mathbf{o}}[k+1] - 1$.

4. For position indices $1 \leq k < N$ in $\mathbf{o}^1$, determine the maximum difference between the position index and its corresponding eigenvalue. i.e. calculate $M = \max\left\{k - f^{\lambda}_{\mathbf{o}^1}[k]\right\}, 1 \leq k < N$.

5. Form a set of search sub-sequences consisting of the following proper prefixes of $\mathbf{o}$: $P = \{\mathbf{o}_{1,k'-1}, \mathbf{o}_{1,k'}, \mathbf{o}_{1,k'+1}, \ldots, \mathbf{o}_{1,M}\}$.

6. Define $k_b = 1$ and $k_e = N - 1$.

7. From each sub-sequence $\mathbf{p} \in P$ (in order from shortest to longest), search $\mathbf{o}^1$ for where $\mathbf{p}$ *first* occurs, searching from position index k_b through to position index k_e. If the search is *successful*:

   (a) Let the position index corresponding to the *first* symbol of the found sub-sequence be k_f.

   (b) At the position index k'' corresponding to the *last* symbol of the located sub-sequence, increment the eigenfunction: $f^{\lambda}_{\mathbf{o}^1}[k''] = f^{\lambda}_{\mathbf{o}^1}[k''] + 1$.

   (c) Set $k_b = k_f$, and repeat step 7 for the next sub-sequence $\mathbf{p} \in P$.

   If the search is unsuccessful, proceed immediately to step 8.

8. Evaluate the final eigenfunction value $f^{\lambda}_{\mathbf{o}^1}[N]$, for $\mathbf{o}^1$, via whatever 'usual' technique is used (to evaluate $f^{\lambda}_{\mathbf{o}}[k]$ originally).

These steps will be illustrated through an example. Consider a binary occupancy sequence $\mathbf{o} = [\mathbf{0111101111101010}]$, with length $N = 16$. It can be shown that its eigenfunction is given by:

$\mathbf{o} =$	0	1	1	1	1	0	1	1	1	1	1	0	1	0	1	0
$k =$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$f^{\lambda}_{\mathbf{o}}[k] =$	1	2	2	2	2	5	5	5	5	5	7	7	7	12	12	12

Suppose that a single-step update is applied to this sequence, resulting in a new sequence $\mathbf{o}^1 = [\mathbf{1111011111010101}]$. It can be shown that its eigenfunction is given by:

$$\begin{array}{rcccccccccccccccc}
\mathbf{o}^1 = & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & \ldots \\
k = & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & \ldots \\
f^{\lambda}_{\mathbf{o}^1}[k] = & 1 & 1 & 1 & 1 & 5 & 5 & 5 & 5 & 5 & 6 & 6 & 6 & 11 & 11 & 11 & \ldots
\end{array}$$

(Note that position $k = 16$ is neglecte,d as the eigenfunction value at that position is evaluated via the 'usual' approaches, not the 'fast' approach described here.)

The above steps are applied as follows:

1. With respect to $\mathbf{o}$, the first symbol is 0, and the first symbol that isn't a 0 occurs at position $k = 2$. So $k' = 2$.

2. With respect to $\mathbf{o}^1$, the value of the eigenfunction is set to 1 for position indices $1 < k < 1$. Clearly no value of k satisfies this, so there is nothing more to be done for this step.

3. With respect to $\mathbf{o}^1$, for position indices $1 \leq k < 16$, set $f^{\lambda}_{\mathbf{o}^1}[k] = f^{\lambda}_{\mathbf{o}}[k+1] - 1$ (one less than the eigenfunction values for the corresponding positions in $\mathbf{o}$):

$$\begin{array}{rcccccccccccccccc}
\mathbf{o}^1 = & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & \ldots \\
k = & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & \ldots \\
f^{\lambda}_{\mathbf{o}^1}[k] = & 1 & 1 & 1 & 1 & 4 & 4 & 4 & 4 & 4 & 6 & 6 & 6 & 11 & 11 & 11 & \ldots
\end{array}$$

4. For position indices $1 \leq k < 16$ in $\mathbf{o}^1$, the maximum difference between the position index and its corresponding eigenfunction value is $M = 5$, occurring at position index $k = 9$ $(9 - 4 = 5)$.

5. The set of sub-sequences to search for is hence:

$$\begin{aligned}
P &= \{\mathbf{o}_{1,1}, \mathbf{o}_{1,2}, \mathbf{o}_{1,3}, \mathbf{o}_{1,4}, \mathbf{o}_{1,5}\} \\
&= \{0, 01, 011, 0111, 01111\}
\end{aligned}$$

6. $k_b = 1$ and $k_e = 15$.

7. Start with sub-sequence $\mathbf{p} = 0$, search for it in $\mathbf{o}^1$, starting from position 1. It is found at position $k = 5$, with the last symbol position at $k'' = 5$. So increment $f^{\lambda}_{\mathbf{o}^1}[5]$ (to $4 + 1 = 5$), and now $k_b = 5$. Now search for sub-sequence 01, starting from position 5. It is found at position $k = 5$, with $k'' = 6$. So increment $f^{\lambda}_{\mathbf{o}^1}[6]$ (to $4 + 1 = 5$), and now $k_b = 5$. Now search for sub-sequence 011, starting from position 5. It is found at position $k = 5$, with $k'' = 7$. So increment $f^{\lambda}_{\mathbf{o}^1}[7]$ (to $4 + 1 = 5$), and now $k_b = 5$. Now search for sub-sequence 0111, starting from position 5. It is found at position $k = 5$, with $k'' = 8$. So increment $f^{\lambda}_{\mathbf{o}^1}[8]$ (to $4+1 = 5$), and now $k_b = 5$. Finally, search for sub-sequence 01111, starting from position 5. It is found at position $k = 5$, with $k'' = 9$. So increment $f^{\lambda}_{\mathbf{o}^1}[9]$ (to $4 + 1 = 5$). The result is now:

$$\begin{array}{rcccccccccccccccc}
\mathbf{o}^1 = & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & \ldots \\
k = & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & \ldots \\
f^{\lambda}_{\mathbf{o}^1}[k] = & 1 & 1 & 1 & 1 & 5 & 5 & 5 & 5 & 5 & 6 & 6 & 6 & 11 & 11 & 11 & \ldots
\end{array}$$

   which matches what we expected.

8. $f^{\lambda}_{\mathbf{o}^1}[16]$ is calculated the 'usual' way – the same way that $f^{\lambda}_{\mathbf{o}}[16]$ was originally calculated.

The algorithm described in this section for calculating the LZC of a sequence, subject to a single-step update, was implemented as a MATLAB function, and its execution speed was *profiled*. This was compared to the execution speed of a 'reference' MATLAB function which calculated the LZC of the sequence the 'naive' way based on the theories described in [43] and evaluated the eigenfunction from scratch. This reference function was written by the author of this thesis and submitted to the MATLAB File Exchange service, which allows authors to share MATLAB code with the wider community. Submissions are reviewed and approved by MathWorks, and are covered by the BSD License. The function, called *calc_lz_complexity*, is available for download – see [89].

The MATLAB Profiler indicated that a reduction in the number of processor clock

cycles by a factor of 8.5 to 12 was possible using this approach. Values at the lower end of the scale were observed with high-complexity sequences, as well as ones with a high probability of bit inversion at each epoch. Values at the higher end of the scale were observed with low-complexity sequences. These results were observed with a sequence length of $N = 256$, on a desktop computer with four Intel Core i5-750 processors clocked at 2.67 GHz, running MATLAB version R2011b.

In summary, the eigenfunction of an occupancy sequence is calculated in the first instance, using some suitable approach based on the definitions of LZC described in [43]. After this, as long as the sequence is only subjected to a single-step update, the eigenfunction of the modified sequence can be evaluated with less computation than would be required when calculating it the initial, 'naive' way. The trade-off is an increase in the computer *memory* requirement, as the complete eigenfunction associated with the previous LZC evaluation must be stored. After the eigenfunction is calculated, the exhaustive complexity can then be calculated normally via (5.11) and (5.12).

5.5 Evaluating Channel Selection Policies

Section 5.3 discussed how a cognitive radio could learn a policy for selecting channels to use, based on characterizing the DC and LZC values of their licensed network user's occupancy patterns. Computing LZC is more computationally demanding than DC, and in section 5.4, a low-computational-complexity approach for continually calculating the LZC of an occupancy pattern was presented. The ability to perform this calculation continually has the pleasant side-effect of catering for a change in the LZC value of a frequency band, due to factors such as a change in location or sensitivity of the licensed network user's traffic patterns to time of day, in a graceful manner.

In this section, these two outcomes are integrated together. Having learned a channel selection policy that is optimal, and given the capability to evaluate the LZC of a channel, a cognitive radio can now apply the policy in dynamic spectrum access. MATLAB simulations were set up to which modelled a cognitive radio doing exactly

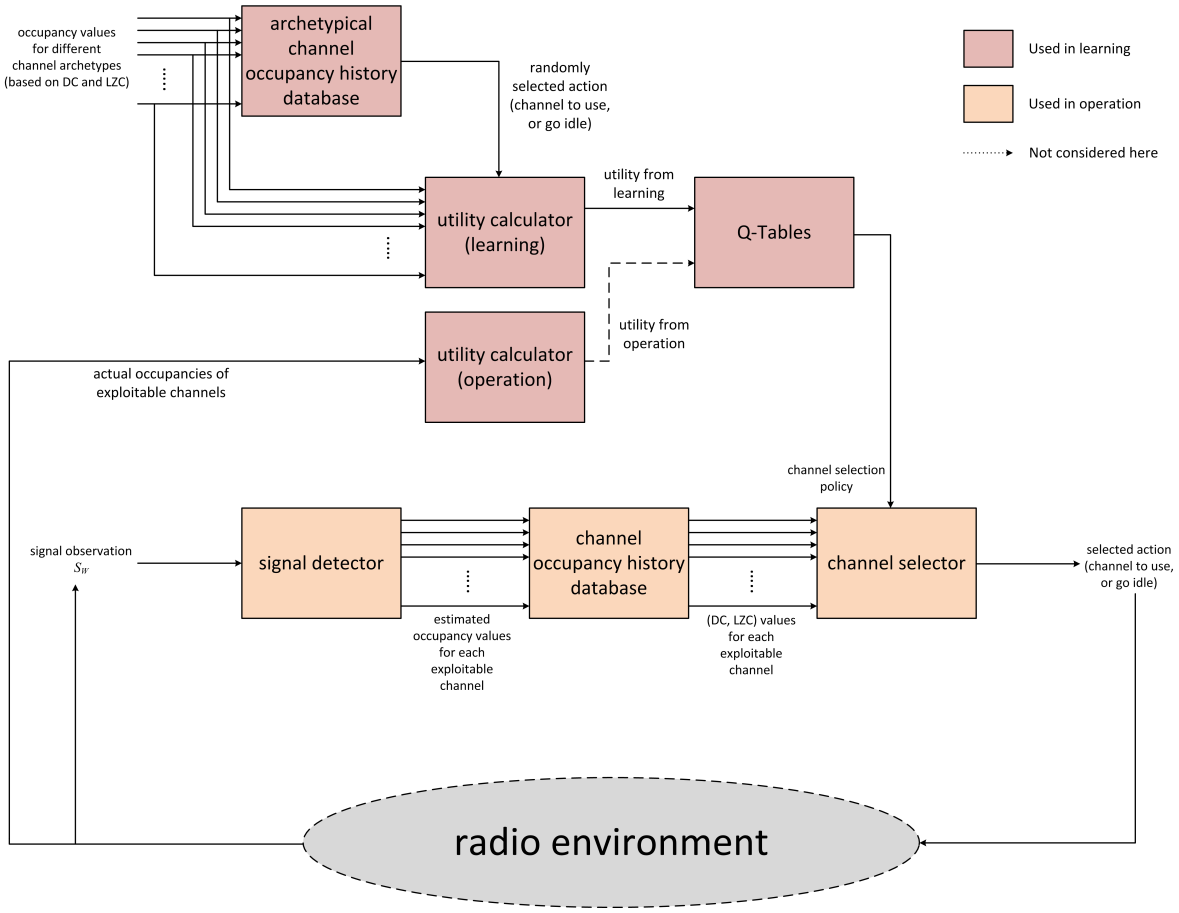

Figure 5.5: A model of a cognitive radio's application of a channel selection policy in operation, mimicked in a MATLAB simulation. The learning blocks are a simplified representation of Figure 5.3.

this. A diagram of the learning and operational processes that are represented in the simulation can be found in Figure 5.5.

In this simulation it is assumed that *the learning of a channel selection policy* and *its application in dynamic spectrum access* are two separate processes with no feedback. In other words, once the policy is learned, it is static and not adjusted based on feedback from experience gained in operation. Conceptually, it is possible to 'close the loop' (via the dashed arrow in Figure 5.5) so that experience obtained during operation is fed back to the learning process to possibly improve the channel selection policy. This matter is discussed further in section 5.6.

The main components and *assumptions* of the simulation are as follows:

- The cognitive radio is presented with a radio environment of N_c *licensed network user channels* which, together, comprise the total *sensing bandwidth* (which was explained in Chapter 4).

- The sensing bandwidth is comprised of K frequency bins.

- Each licensed network user channel has a fixed bandwidth, represented by $\frac{K}{N_c}$ frequency bins.

- The licensed network user transmitters are *homogeneous*, in the sense that they all have the same bandwidth requirement, and all employ the same modulation scheme.

- The cognitive radio is assumed to have a fixed bandwidth requirement, represented by B frequency bins. For the purposes of this exercise, it is assumed that it is not able to dynamically change its bandwidth requirement by employing adaptive modulation techniques. Furthermore, the B frequency bins used are assumed to be *contiguous*.

- Every possible set of B contiguous frequency bins within the sensing bandwidth represents a channel that may be exploited by the cognitive radio so, from its perspective, there are $K - B + 1$ exploitable channels.

- The cognitive radio determines the DC and LZC values based on the result of a spectrum sensing operation. For realistic spectrum sensing schemes, the occupancy of each frequency bin as determined by the cognitive radio (the *sensed occupancy*) may be different from the *actual occupancy*, due to signal detection error.

- An exploitable channel is deemed to be occupied by the cognitive radio at time t_n if *any* of its frequency bins are considered occupied at time t_n, as indicated by its spectrum sensing operation. In other words, if the sensed occupancy of frequency bin k at time t_n is denoted by $o_k[n]$, then the sensed occupancy of

exploitable channel c at time t_n is denoted by $O_c[n]$:

$$O_c[n] = o_c[n] \vee o_{c+1}[n] \vee o_{c+2}[n] \vee \cdots \vee o_{c+B-1}[n], \tag{5.49}$$

where $\vee$ denotes the logical OR operation.

- The simulation knows the *actual* occupancy patterns of each licensed network user's transmission, and hence the actual occupancy of each frequency bin. It is hence able to evaluate (5.49) to determine the actual occupancy of each exploitable channel.

It is worth noting that in the simulation, the cognitive radio evaluates DC and LZC based on the *sensed* occupancy of each exploitable channel, whereas the immediate utility at each epoch, r_t (see (5.38)), is evaluated based on the *actual* occupancy of each exploitable channel. If perfect spectrum sensing and signal detection is possible, then sensed occupancy is the same as actual occupancy.

One dilemma that needs to be resolved at this point is – how should the simulated radio environments be constructed? There are two facets to this problem:

- How are channel *archetypes* distributed with respect to frequency? Different channel archetypes have been discussed earlier in this chapter. Should each channel's archetype be selected randomly and independently of each other? Or should a more sophisticated distribution of archetypes between channels be undertaken?

- Within each channel, how are the *burst* times (when the licensed network user is transmitting) and *run* times (when the licensed network user is idle) determined? Are they determined randomly with weighting provided by value of DC? Or again, is something more sophisticated warranted?

It would be possible to simply employ the same corpus of spectrum measurements obtained from RWTH Aachen University [87], used to supply channel occupancies which met certain DC and LZC archetypical requirements during the Q-learning process. However, as this occupancy source was used during training, it was decided that

an alternative source of channel occupancies would be used for these simulations, in order to support a more generalized and independent verification of the efficacy of the policy learning process. It turns out that [90] provides answers to these questions.

When they analyzed the real-life spectrum occupancy data obtained from [87], they found that the burst and run lengths of a transmitter could be modelled with reasonable fidelity using either a *lognormal* or *geometric* distribution. The parameters of these distributions varied depending on:

- How 'busy' the transmitter was;
- Which spectrum band was being examined (one of: Global System for Mobile Communications (GSM) 900 MHz uplink, GSM 1800 MHz downlink, Digital Enhanced Cordless Telecommunications (DECT), ISM 2.4 GHz);
- The location in [87] where the data was collected.

For a given location and spectrum band, they classified each channel according to the following *spectrum-use archetypes*, describing how 'busy' the transmitter is, which loosely corresponds with its DC measure:

- Vacant (never occupied)
- Low
- Mid[4]
- High
- Full (always occupied)

For each spectrum-use archetype, a statistical distribution (either lognormal or geometric) was provided for the burst and run length – implying that the burst and run lengths can be modelled as iid random variables.

Furthermore, they found that for a given location and spectrum band:

[4]The term 'mid' is used by the authors in [90], which approximately corresponds with the term 'medium' used in this thesis to describe DC.

	Radio Environment 1	**Radio Environment 2**
Location	Aachen, Germany; third-floor balcony in a residential area	Maastrict, Netherlands; school rooftop in a residential area
Spectrum Band	GSM 900 MHz uplink	GSM 900 MHz uplink

Table 5.4: Radio environments whose temporal and spectral characteristics are synthesized based on models in [90] to determine the efficacy of learned channel selection policies.

- The distribution of channels between spectrum-use archetypes is non-uniform;
- Channels belonging to the same spectrum-use archetype tend to be 'clustered' together in frequency, and the number of channels in each cluster could be modelled with reasonable fidelity using a lognormal distribution.

Hence, they also provide lognormal distribution parameters for the spectrum-use archetype cluster sizes, as well as a procedure for allocating channels to spectrum-use archetypes when attempting to construct a radio environment to mimic a particular spectrum band and location (see [90] – *Appendix. Model Parameters*).

The simulations in this section implement this procedure, as well as all of the aforementioned statistical models, to mimic two kinds of radio environments, as described in Table 5.4.

Once a radio environment has been created, the simulation models the decision process of a cognitive radio performing dynamic spectrum access:

1. The cognitive radio begins in an idle state.
2. The cognitive radio *estimates* the frequency bin occupancies across the sensing bandwidth.
3. The cognitive radio then selects a channel based on its learned channel selection policy, considering actions from most preferred to least preferred in turn, depending on whether the given action is possible in the current channel state of the radio environment or not.
4. Simulation time is then advanced by one epoch.

5. The *actual* licensed network user occupancies of the radio environment are determined for each frequency bin across the sensing bandwidth, based on whether or not each licensed network user is transmitting or not at the new epoch.

6. The immediate utility r_t for the new epoch is evaluated, based on the action selected by the cognitive radio in the last epoch, and the actual occupancies of each frequency bin by the licensed network users. r_t is added to an *accumulator* of the total utility encountered over the simulation since it began. *r_t is calculated in the same way, with the same parameters, as during Q-learning from which the channel selection policy was derived.*

7. Steps 2 - 6 are repeated in a loop, for a specified number of epochs.

From the point of view of the cognitive radio, *any* B contiguous frequency bins comprises an exploitable channel. Hence, the cognitive radio's spectral occupancy is independent of the channelization (channel boundaries) of the licensed network users. *This is a direct application of the dynamic spectrum access principles subject to arbitrary channelizations described in Chapter 4.*

Some other simulation parameters are listed in TABLE 5.5.

In order to demonstrate the efficacy of the learned channel selection policy, the simulation must also be run with the cognitive radio employing a 'less-informed' channel selection policy as a reference. In this case, a *random policy* is used as a reference, described as follows:

- Pick with equal probability *any* channel that appears to be vacant
- Continue utilizing the selected channel until it appears to be occupied, then switch with equal probability to any other channel that appears to be vacant
- Only go into an idle (quiescent) state and suspend transmission if there is *no* channel that appears to be vacant

TABLE 5.6 shows simulation results where perfect spectrum sensing is assumed – there is no discrepancy between actual spectral occupancies and those seen by the

Total no. radio environments simulated	48
Total no. epochs simulated per radio environment	10000
No. licensed network user channels in radio environment	256
No. frequency bins per licensed network user channel	16
Total no. frequency bins in sensing bandwidth (K)	$256 \times 16 = 4096$
No. contiguous frequency bins needed by cognitive radio (B)	32
Licensed network user signal modulation	OFDM with symbol length 1024; guard interval 1/8 of a symbol; data sub-carriers QPSK modulated; pilot sub-carriers with amplitude 8/3

Table 5.5: Parameters for simulation to verify the benefit of employing a learned channel selection policy in operation.

cognitive radio. It shows that there is a significant improvement in the average net utility (benefits discounted by costs) to be gained from employing a learned channel selection policy.

Some other notable observations from these simulation results are:

- The average benefit associated with transmitting, B_c, is slightly less when applying the learned channel selection policy, because the learning process has identified that there is a long-term benefit associated with going idle in some circumstances, even if there are vacant channels, so the cognitive radio spends more time overall in an idle state. On the other hand, the random policy only allows for this when there are no vacant channels.

- For *Radio Environment 1*, very little cost is expended in channel switches for both the random and learned policies – but for somewhat different reasons. *Radio Environment 1* has a high proportion of low-use channels (72%), as well as a small proportion of completely vacant channels. With the learned policy, channels with

		Radio Environment 1		Radio Environment 2	
		Random Policy	**Learned Policy**	**Random Policy**	**Learned Policy**
Utility Function Parameters (see Table 5.1)		Case 2	Case 2	Case 1	Case 1
Average Benefit per Epoch (see (5.38))	B_c	5.000	4.808	5.000	4.011
	$C_{i,out}$	-5.992	-0.098	-16.019	-6.366
	$C_{i,in}$	-0.300	-0.005	-1.602	-0.637
	C_s	-0.060	-0.001	-0.320	-0.170
Net Utility per Epoch (see (5.38))	r_t	-1.352	4.704	-12.941	-3.162

TABLE 5.6: Average reward received per epoch for a cognitive radio employing different channel selection policies. Negative values indicate 'costs' (or 'negative benefits'). Perfect (error-free) spectrum sensing is assumed (i.e. sensed occupancy is the same as actual occupancy).

low DC, low LZC are favoured, causing the cognitive radio to naturally gravitate towards utilizing one of the low-use or vacant channels. If it happens to pick a vacant channel, it never has to switch ever again! Even if it picks a low-use channel, it will not have to switch channels often and, when it does, there is a high probability of another low-use channel being available. With the random policy, the high proportion of low-use channels simply means that they will be selected most of the time anyway.

- For both the random and learned policies, significantly more costs are incurred associated with interference to and from the licensed network users in *Radio Environment 2* compared to *Radio Environment 1*. *Radio Environment 2* has no vacant channels and a relatively even distribution between low-, mid- and high-use channels. For the learned policy, it turns out that most of the channels that the cognitive radio ends up using are those with low DC, medium LZC (the third preference in most circumstances); low DC, low LZC and medium DC, low LZC opportunities (higher preference) rarely occur. For the random policy, the higher

prevalence of higher-occupancy channels means more interference and switching costs overall.

Table 5.7 shows simulation results when imperfect spectrum sensing (a more realistic scenario) is employed (for Radio Environment 2 only) – there may be discrepancies between actual spectral occupancies and those seen by the cognitive radio. The main additional characteristics of the simulation in this case are:

- At each epoch, the simulation knows which licensed network users are transmitting and which ones are not. With the signal modulation parameters of each licensed network user defined, the cognitive radio is able to obtain a PSD estimate, $S[k]$ across the sensing band.

- With the PSD estimate, the cognitive radio is able to use the peak detection technique based on maximal slope magnitudes (refer to section 4.2) to isolate a spectral 'chunk' possibly associated with a licensed network user's transmission. By then employing the decision process described in section 4.1.2, the occupancy of that spectral chunk is estimated. By employing an iterative method similar to that described in Algorithm 4.1, the cognitive radio estimates the occupancy of every frequency bin across the sensing band.

- In applying the decision process described in section 4.1.2, a detection threshold λ_T was empirically determined so that, for the PSD estimate of a licensed network user's transmission, the average PSD value in a frequency bin is below this threshold for approximately 27.5% of the time. This threshold is fairly arbitrary, but it meant that the cognitive radio did not see too many frequency bins being falsely alarmed due to the noisy environment, which would cause the cognitive radio to go into an idle state for the vast majority of the time.

- The simulation sets up a radio environment with SNR = 0 dB and AWGN. There is no channel fading.

It is apparent that, even with the imperfect spectrum sensing, and with the given utility function being considered, a significant improvement in overall return is obtained

		Radio Environment 2	
		Random Policy	**Learned Policy**
Utility Function Parameters (see Table 5.1)		Case 1	Case 1
Average Benefit per Epoch (see (5.38))	B_c	5.000	4.343
	$C_{i,out}$	-35.382	-19.629
	$C_{i,in}$	-3.538	-1.962
	C_s	-0.318	-0.109
Net Utility per Epoch (see (5.38))	r	-34.238	-17.357

TABLE 5.7: Average reward received per epoch for a cognitive radio employing different channel selection policies. Negative values indicate 'costs' (or 'negative benefits'). Imperfect spectrum sensing was employed, with the spectral occupancy of licensed network user transmissions estimated according to the scheme described in section 4.1.2.

when employing the learned channel selection policy.

5.6 Final Remarks – Continually Learning from Experience

In this work, the processes of learning and applying a channel selection policy are treated as rather separate, and the latter does not feed back into the former. The learning process described in section 5.3.3 was carried out 'offline' using synthesized traffic occupancy sequences known to match one of the six channel archetypes of interest. This means that the instant utility, r_t, could be computed exactly because the *actual* channel occupancies were known.

Referring to FIGURE 5.5, if it were possible to similarly determine the instant utility at each epoch when applying the learned policy, then the Q-tables upon which the channel selection policy is based could be continually updated. Learning then becomes a continuous process which could result in an improvement in the policy even whilst the cognitive radio is operating (via the dashed arrow in FIGURE 5.5). The problem is that a cognitive radio in the field cannot know the exact state in which a

channel selection action is taken because it can only *estimate* the occupancies of the channels in the radio environment using imperfect signal detection. This means that at each epoch, *both the instant utility, and the entry in the Q-tables which is updated, are subject to error*[5].

It would be interesting to investigate this matter further. One could see if feeding back 'approximate' utility calculations (based on what the cognitive radio 'sees', which may be different from reality) and updating Q-tables based on 'approximate' state results in any average utility gain over time, compared to employing a static channel selection policy. If this is found to be the case, one could envisage a cognitive radio being pre-loaded with a reasonably useful policy obtained from offline learning, and then continually refining it based on experience gained in the field.

Further concluding remarks for this chapter now follow.

5.7 Conclusion

This chapter has considered concepts from several fields of knowledge in an integrative manner to address a fundamental question: how does a cognitive radio select which channel to utilize when there are many options available?

It was noted that different channels have different licensed network user occupancy characteristics – at the most simplistic level, a channel that is vacant most of the time will provide more benefit to a cognitive radio performing dynamic spectrum access than one that is hardly ever vacant. Even on this measure, it makes sense that the former should be preferred. The DC measures the overall proportion of time that the channel is occupied. Further discussion is based on the premise that DC is a one-dimensional measure of the occupancy characteristics of a channel – employing LZC, which gives an indication of the unpredictability of the occupancy pattern, in addition to DC enriches the available information with which channel selection decisions can be made.

Machine learning techniques can then learn an optimal channel selection policy from these channel occupancy characteristics. Q-learning was employed – it does not require

[5] In a contrived simulation like that described in section 5.5, the exact occupancies of the channels are known. However, in this work this is only used to calculate the average utility per epoch, not for continuous learning.

any pre-conceived notion of the radio environment, and it allows for highly exploratory behaviour during learning to consider the utility resulting from all possible states and actions. The benefit of developing a channel selection policy in this way was verified using simulations, compared to a more naive policy of simply selecting a vacant channel at random. The channel selection policies obtained in this chapter corroborated the suggestion that there is a benefit in considering LZC in addition to DC – it was shown that LZC is actually a stronger indicator of how favourable a channel is for use by a cognitive radio compared to DC, with a medium DC, low LZC channel resulting in better utility than a low DC, medium LZC channel for the scenarios considered in this chapter.

Some remarks in relation to the Q-learning that was carried out in this work, as described in section 5.3.3 and TABLE 5.1, are warranted at this point. The learning rate used results in relatively gentle updates to the Q-table in each epoch, and the discount factor favours long-term over short-term utility.

The emphasis on long-term utility implies that the benefit available to the cognitive radio can be deferred into the future. Therefore, the resultant channel selection policy may allow for short-term costs to be incurred if it is likely to lead to long term gain. For example, the policy may suggest that a channel switch takes place, even though the channel currently being used is still vacant. The extent to which utility can be deferred in the long term may depend the delay sensitivity of the traffic to be carried over the cognitive radio network. One may find it interesting to consider the effect of the discount factor on the learned channel selection policy, and the benefit of applying each policy to traffic with different latency requirements.

What was not mentioned previously was the time taken for the Q-tables to converge, at which point learning ceased. In these simulations, the number of elapsed epochs before convergence varied from about 50000 to 50000000 (depending on the pair of channel archetypes being considered). Whilst long convergence times are not so much of a problem for offline simulations, they are detrimental to performing learning in the field. Any measure that can reduce the convergence time would be tremendously valuable to this work. The value of the learning rate may influence the convergence

time.

In order to employ LZC in channel selection, it must be possible to calculate this efficiently. A fast LZC calculation method was presented, which works on the presupposition that a cognitive radio may want to recalculate the LZC of a channel's occupancy sequence each epoch. The method concentrates on efficiently calculating the sequence's eigenfunction, and operates on the principle that, if the eigenfunction of a sequence has been calculated, and if the sequence is then only changed by one bit and a left shift at each epoch (a single-step update), then most of the computational effort expended in calculating the previous sequence can be preserved, with the effect of the left shift considered as an adjustment to the previous eigenfunction to obtain the new one. Whilst it may appear extreme to evaluate LZC each epoch, this has the benefit of not having to worry about *how often* it needs to be re-evaluated, as changes to a channel's LZC characteristics are gracefully taken into account as the new occupancy data propagates through the channel's history buffer.

An observation that must be made is that occupancy sequences are a *discretized* representation of a channel occupancy. After all, a channel's occupancy is defined for all time, so it can be treated as a *continuous* function of time (with value of either 0 or 1 at a given time). There are several implications of this which have not been addressed in this chapter.

The 'samples' of the occupancy sequence have a 'sample period' equal to the duration between epochs in the decision process. Throughout the discussion in this chapter, this time period has been not been explicitly stated; a nominal value has been presumed. In reality, this is likely to be an important parameter. If the period is too small, then too much power is consumed in spectrum sensing operations, and a large data transmission opportunity cost is incurred. On the other hand, if the period is too large, the cognitive radio network becomes unresponsive to change, and a large cost can be incurred due to interference with licensed network users. Furthermore, the absolute values of LZC will have a dependence on this time period. It would seem apparent that a shorter time between epochs will cause a general downward shift in all LZC values – occupancy sequences will tend to appear less complex, since more

occupancy samples will be taken in each burst and run, giving the appearance of a more slowly varying sequence. Whilst the qualitative classification of LZC values into 'low', 'medium' and 'high' suggests that the relative values are more important than the absolute values anyway, a suitable range of LZC still needs to be selected for each class. Clearly, the selection of the epochal time period has wide implications, and ways of selecting appropriate values for it warrant further investigation.

The time period between epochs is also implicated in some of the shortcomings in the formulation of the utility function presented in this chapter. The terms representing the cost of interference are considered to be 'binary' in nature:

- It is assumed that if a 'collision' occurs at an epoch (where the channel is found to be occupied by the licensed network user whilst the cognitive radio is using it), that interference has occurred for the *entire* time period since the last epoch. This is a pessimistic assumption – it is possible that the licensed network user only very recently started transmitting, and that the majority of that time period was actually free of interference. However, *the cognitive radio has no knowledge of what happened in between epochs*, and so cannot make this determination.

- No consideration is made with respect to the relative locations of the cognitive radio transmitter and the receiver terminals on the licensed network. It is assumed that, if the channel is seen by the cognitive radio to be occupied by a licensed network user, then any attempt to use the channel will result in the full cost of interference being incurred. The reality is that the relative locations and transmission directionality may mean that only a fraction (or even none) of this cost is actually incurred. Conversely, it is also assumed that, if the channel is seen by the cognitive radio to be vacant, then it will get the full benefit of channel use by transmitting with zero interference cost – again, in reality, the interference cost may be somewhere between zero and the maximum. These observations are fundamentally symptoms of the *exposed-* and *hidden-node* problems respectively, which were described in section 1.2.1.

The channel selection policy learning outcome is dependent on how the form of the

utility function, so a more sophisticated utility function may result in better outcomes in reality. It is possible that the concepts described in this chapter could be further considered in the context of cooperative spectrum sensing to address these matters.

6

Conclusion

The preceding chapters of this dissertation have covered several topics addressing fundamental problems relating to software-defined radios and cognitive radios performing dynamic spectrum access. The research work that has been undertaken since Mitola's vision of 'smart' wireless communication devices was elucidated to the wider community over a decade ago is rich and varied. Much like, and possibly motivated by, the demand for ubiquitous wireless services to communicate and entertain, the research continues to burgeon. There is a recognition that changes are required in how wireless networks are established and spectrum is provisioned to support them in order to increase service capacity in the future, and this should happen sooner rather than later.

It has not been possible to address all of the problems which remain to be solved in order for wireless communication devices to become fully flexible, self-organizing and

as spectrum-efficient as possible in this thesis. The issues that still warrant attention, and the research directions that could be taken from the status quo, are myriad. This thesis has been necessarily constrained in its scope. In particular:

- The main concern for cognitive radios that is considered here is *dynamic spectrum access* – achieving an increase in overall spectrum utilization by allowing devices to dynamically recognize and exploit vacant or under-utilized channels. Other ways in which cognitive radios can flexibly change their operational parameters such as transmission power, adaptive modulation, channel coding or transmission directionality are not considered.

- Attention is largely restricted to the *physical* layer of the communications protocol stack, with some elements of media-access control as well.

Particularly on the topic of dynamic spectrum access, this thesis makes contributions on the following matters:

- How a *single* device can recognize whether or not a channel is occupied or not by another transmitter, by analyzing a signal observation that it takes from the radio environment, balancing the competing motivations of *reliable determination* and *low power requirement*.

- Given a selection of many vacant channels that it could use for its communication needs, how a device should select which one to use based upon the observations that it makes.

The remainder of this chapter will summarize the main outcomes and achievements which have arisen from this thesis, and then proceed to suggest how these could be enriched through further research.

6.1 Recapitulation of the Major Outcomes of this Thesis

6.1.1 Recognizing Vacant and Partially-Vacant Channels

A cognitive radio performing dynamic spectrum access fundamentally needs to reliably recognize if a channel within some frequency band is occupied or not by another wireless transmitter. To do this, it must take a radio-signal observation of the radio environment and use the available evidence from the observation to make this decision. The decision is generally based on some form of signal detection algorithm.

In this respect, the thesis has made the following contributions:

- Whilst cyclostationary detection is often cited as a distinct approach to energy detection, the principle that cyclostationarity detection is actually a *superset* of energy detection is established. Cyclostationarity detection allows for the consideration of spectral *features* in a modulated signal beyond, and including, its power spectral density. It is the consideration of these additional spectral features that allows cyclostationarity detection to achieve better detection-error performance than energy detection. (Chapter 2)

- Cyclostationarity detection is advocated for spectrum sensing and signal detection because of its ability to outperform energy detection in terms of detection reliability. However, calculating the estimate of a signal's spectral correlation density (SCD) is costly in the amount of computation required, and hence the amount of power consumed. The principle of drastically reducing the amount of computation required by only considering certain relevant regions on the $f - \alpha$ plane is established. The assumption that the entire SCD calculation is required to implement a cyclostationarity detector is challenged. This principle effectively removes redundant calculations from the cyclostationarity detector. (Chapter 2)

- Accordingly, it was shown that SCD estimation algorithms such as the FFT Accumulation Method can be used to perform an SCD estimate on any arbitrary

subset of the $f-\alpha$ plane that is desirable for cyclostationarity detection. (Chapter 2)

- The regions on the $f-\alpha$ plane over which a cyclostationarity detector should perform its SCD calculation will vary depending on the particular modulation parameters employed by other transmitting devices of interest. It was shown that a detector can *learn* the regions that are relevant by considering a set of observations where it can be established with a high degree of certainty that the signal is in fact present, and these observations may be taken by the device out in the field (Chapter 2, Chapter 3)

- Furthermore, a trade-off between the time spent in learning the relevant regions on the $f-\alpha$ plane and the resulting detector performance is identified. It was mentioned in the previous dot point that learning takes place on a set of observations where there is a certain degree of certainty that they capture the signal of interest. Increasing the degree of certainty requires an increase in the time spent learning, but the resulting detector will have better detection-error performance. The existing literature on cyclostationarity detection for cognitive radios does not seem to address or even mention practical considerations such as these – presumably because they presuppose that a complete SCD estimate on the entire $f-\alpha$ plane is required. (Chapter 3)

- A comparison of the amount of computation (and hence power) required between cyclostationarity detection and energy detection was presented, highlighting the magnitude of the saving that is possible by considering only specially selected regions on the $f-\alpha$ plane. (Chapter 2)

- The oft-made assumption, that devices which a cognitive radio is required to detect the presence of will transmit according to known centre-frequencies and bandwidth, is challenged. A generalized energy detector that relaxes this assumption is presented – one that makes a decision, not only on whether or not a transmitted signal is present in a channel, but also on the frequency range

being utilized by a transmitted signal. This opens up the traditional scope of the spectrum sensing problem, paving the way for cognitive radios to sense *each other's presence* as well as that of licensed network users – something that gets comparatively scant attention in the research literature. (Chapter 4)

6.1.2 Selecting the Most Beneficial Channel to Use

Having identified what channels are available to be used by a cognitive radio, it must then rank or prioritize the available opportunities based on which one it believes will provide the most benefit. There are two ingredients that are needed to make this possible which are considered in this thesis: the channels need to be characterized in a suitable way, and then a method to determine a suitable policy for selecting channels based on these characterizations is required.

In these areas, this thesis makes the following contributions:

- It is demonstrated that simply considering all channels that are vacant as offering the same 'quality' of opportunity to a cognitive radio is naive. In addition, consideration of the duty cycle of a channel (the fraction of total time that it is occupied) as the primary characterization upon which to base channel selection is also inadequate. Despite this, the duty cycle is the quality that receives the most attention in the research literature. This work builds upon the work of others that considers the unpredictability (*complexity*) of the occupancy pattern associated with a channel in addition to the duty cycle. An observation was made, at least in some cases, that a channel that is occupied more often but whose occupancy is more predictable (less complex) can provide more benefit than one that is occupied less but whose occupancy is less predictable (more complex). This challenges the approach made in much of the literature in relying solely on the duty cycle, which would suggest simply selecting channels that have low occupancy. It also strongly advocates the need for cognitive radios to characterize channels in a more sophisticated way. (Chapter 5)

- LZ complexity is used as the complexity measure for a channel's occupancy pattern in this thesis, and its calculation requires a high degree of computation compared to calculating the duty cycle. This thesis presents an algorithm that can be used to very efficiently compute the LZ complexity of an occupancy sequence of a channel under certain circumstances: the sequence only differs from another sequence by a single value (resulting from a single-step update), and the full eigenfunction of this other sequence is still available. It is believed that this algorithm has not been published elsewhere. (Chapter 5)

- It is possible to use Q-learning to develop a complete channel selection policy that is applicable to radio environments that are heterogeneous (have channels that exhibit a wide range of duty cycle and complexity values in their occupancy), whilst keeping the state- and action-space of the Q-learning problem manageable. Other published work assumes a limited radio environment with relatively few channels and where each channel has the same occupancy characteristics, resulting in a policy that by itself has limited utility. By amalgamating learning outcomes from many such constrained environments, a more useful channel selection policy applicable to radio environments that are hetereogeneous and even time-varying can be constructed. (Chapter 5)

- The benefit of employing an 'informed' channel selection policy from machine learning is demonstrated even when a cognitive radio does not have complete channel state information, and must therefore resort to one of the practical signal detection algorithms such as energy detection or cyclostationarity detection to estimate the occupancies of each channel over time. Contrastingly, published work in this area tends to simply assume that the cognitive radio is able to perform signal detection with perfect accuracy. (Chapter 5)

6.2 Where to from Here?

Just as this thesis built upon the research work of others, it is hoped that this dissertation itself could inspire new work to be carried out.

It is apparent with some thought that there will be a significant benefit in incorporating *cooperative* approaches into the work presented in this thesis. For example, the work considers the temporal (time) and spectral (frequency) dimensions of exploiting vacant or under-utilized channels, but not the spatial dimension (see section 1.1.3). It has already been shown that spatial considerations and directionality in the transmissions of cognitive radios and licensed network users need to be known in order to solve the exposed-node and hidden-node problems (see section 1.2.1). Clearly, the spectrum sensing or signal detection result of a single cognitive radio terminal does not encapsulate sufficient information to avoid these problems.

Conceptually however, one could envisage a cognitive radio *infrastructure*, consisting of some form of spectrum brokering service, that is able to decide if it is 'safe' for a cognitive radio transmitter to transmit to its intended receiver, by using spectrum sensing information from geographically diverse terminals. This decision is made on the basis of additional information that the cognitive radio terminal performing spectrum sensing alone would not have. With a sufficient number and diversity of spectrum sensing cognitive radio terminals, the broker would have enough information to estimate the location of a licensed network user *transmitter* which is operating in a given frequency band, as well as the approximate shape of that transmitter's radiation envelope. A cognitive radio, on the other hand, is more interested in the location of the licensed network *receiver* – nonetheless, knowing the approximate shape of the transmitter's radiation envelope would provide some constraints on the location of the corresponding receiver. This discussion assumes that the location of the licensed network user terminals cannot be definitively determined by other means.

In addition, *if the broker also knows the positions of all cognitive radio nodes*, and knows which nodes are in a communication session with each other, then it is in a better position to determine if a cognitive radio transmission will cause interference with any

device operating in the licensed network. This could be enforced by requiring cognitive radio terminals to 'register' with the brokering service when they join the network and when they intend to establish a communication link with another cognitive radio terminal, and report its location periodically.

The concept of a binary occupancy sequence that was described in chapter 5 follows the same conventions as that used in [42], and is consistent with the vast majority of the cognitive radio literature that refers to channels as being 'vacant' or 'occupied' (or semantically equivalent terminology). However, in the context of cooperative spectrum sensing and a spectrum brokering service, the concept of a channel being 'vacant' or 'occupied' *as seen at a particular location* is inadequate.

The main outcomes in this thesis retain their validity in the context of the aforementioned cognitive-radio infrastructure, with a conceptually simple semantic modification to one of the major ideas, described as follows:

- Cognitive radio terminals perform signal detection using the principles in this thesis, and report the result for each frequency band to the brokering service.

- The concept of an occupancy value is simply replaced with that of a *safe-to-transmit* value – 0 means 'safe to transmit', 1 means 'not safe to transmit'. This value is determined by the brokering service for each transmitter-receiver pair in a cognitive radio network.

- Devices on a cognitive radio network can still learn and apply a channel selection policy, but this policy is based on DC and LZ complexity calculated on a sequence of safe-to-transmit values, rather than *locally sensed channel occupancy values*.

Further work could therefore incorporate the ideas in this thesis into the operation of a cognitive-radio infrastructure with the features described above. Another dimension that is then added to the problem is that of protocol messaging and overhead associated with the infrastructure, and *designing protocols that have minimal overhead*.

In Chapter 5, whilst the utility function used proved the value of the concept of a channel selection policy, there is room to improve the fidelity with which it captures the

net overall benefit of communicating over a channel between epochs in a real network. In particular, the following observations can be made:

- The benefits and costs within the utility function are of a *local* nature (i.e. measured for a given cognitive radio link considered in isolation), rather than of a more global (system-wide) nature.

- As mentioned in section 5.7, the amount of interference cost is binary in nature (the entire cost is incurred or none at all), without any scaling based on the fraction of the epoch for which interference actually occurred. This is because the occupancy value (or, as advocated earlier, the safe-to-transmit value) is measured at discrete times, and there is no knowledge of its value between epochs.

- Similarly, the interference cost is not scaled based on the fraction of the bandwidth of the signal that is experiencing interference.

- The utility function results in zero cost and zero benefit (and hence zero net utility) when a cognitive radio transmitter is in an idle state. In reality, *one might expect that there is in fact a cost associated with being idle, and this cost would conceivably increase as consecutive epochs are spent in an idle state.* For example, this could be a *quality-of-service cost* that rises as delay is introduced by consecutive epochs without transmission. There could also be costs associated with *recovery from transmit buffer overflows* that occur once a small number of consecutive epochs in an idle state are encountered.

One could investigate how the utility function used in learning a channel selection policy could be improved, particularly in the context of the aforementioned cognitive-radio infrastructure. The issue of lack of knowledge about when interference starts occurring between epochs could be addressed by 'staggering' the epochs of different cognitive radio links so that spectrum sensing measurements are taken across the system in finer time increments, for example.

Finally, one could consider how learned channel selection policies could co-exist with an internal *database* of learned information based on previous actions. Applying

a channel selection policy at each epoch, as described in chapter 5, is relatively costly from a computation perspective, as it requires spectrum sensing and channel characterization to be re-computed at each epoch for every possible channel (notwithstanding the gains in LZ complexity calculation efficiency that may be possible using the algorithm described). All this computation effort may be eliminated in some circumstances by information that is learned from the past. For example, if past experience indicates that a particular channel is always vacant at a certain location, or at a certain time of day, then when these conditions hold the channel to utilize may be selected based on this, rather than a full channel re-evaluation and selection process. The guiding principle is that *the storage cost associated with maintaining a database of information learned from past experience can be used to reduce the amount of computation associated with channel selection in the future.* Further research is required to advance this idea, including how the database is maintained based on new observations and new learned information, and may result in new frontiers being established in data representation, database design and database optimization (for minimization of memory space and retrieval efficiency).

> "*There is nothing like looking, if you want to find something. You certainly usually find something, if you look, but it is not always quite the something you were after.*" – J. R. R. Tolkien

A

Appendices

A.1 Probability Density Function of λ_{norm}, $R = 0$

Here it is shown that when $R = 0$ (such that $s[n] = s_n[n]$ – see (2.2) and (2.6)), λ_{norm} (as defined in (2.12)) follows the central χ^2-distribution with $2N$ DOF.

In this case, let the total noise power be σ_n^2, which is split equally between the in-phase and quadrature components. Therefore, both $s_I[n], s_Q[n] \sim \mathcal{N}\left(0, \frac{\sigma_n^2}{2}\right)$. It then follows that both $\frac{\sqrt{2}}{\sigma_n} s_I[n], \frac{\sqrt{2}}{\sigma_n} s_Q[n] \sim \mathcal{N}(0, 1)$ (i.e. follows a *standard normal* distribution).

Let $X_0, X_1, \ldots, X_{2N-1} \sim \mathcal{N}(0,1)$ be a sequence of random variables representing $\frac{\sqrt{2}}{\sigma_n} s_I[0], \frac{\sqrt{2}}{\sigma_n} s_Q[0], \frac{\sqrt{2}}{\sigma_n} s_I[1], \frac{\sqrt{2}}{\sigma_n} s_Q[1], \ldots, \frac{\sqrt{2}}{\sigma_n} s_I[N-1], \frac{\sqrt{2}}{\sigma_n} s_Q[N-1]$. From (2.12):

$$\begin{aligned}
\lambda_{norm} &= \frac{2}{\sigma_n^2} \sum_{n=0}^{N-1} |s[n]|^2 \\
&= \frac{2}{\sigma_n^2} \sum_{n=0}^{N-1} s_I^2[n] + s_Q^2[n] \\
&= \sum_{n=0}^{N-1} \left(\frac{\sqrt{2}}{\sigma_n} s_I[n] \right)^2 + \sum_{n=0}^{N-1} \left(\frac{\sqrt{2}}{\sigma_n} s_Q[n] \right)^2 \\
&\equiv \sum_{k=0}^{2N-1} X_k^2, \qquad \text{(A.1)}
\end{aligned}$$

thus, by definition of the central χ^2-distribution, proving the initial assertion.

A.2 Probability Density Function of λ_{norm}, $R = 1$

Here it is shown that when $R = 1$ (such that $s[n] = s_t[n] + s_n[n]$ – see (2.2) and (2.6)), λ_{norm} (as defined in (2.12)) follows the non-central χ^2-distribution with $2N$ DOF with $\delta = 2N \times$SNR.

As in section A.1, let the total noise power be σ_n^2, which is split equally between the in-phase and quadrature components. The $s_t[n]$ component is itself complex, having in-phase and quadrature components:

$$s_t[n] = s_{t,I}[n] + j s_{t,Q}[n]. \qquad \text{(A.2)}$$

At each sample time n, there are two random variables: $s_I[n] \sim \mathcal{N}\left(s_{t,I}[n], \frac{\sigma_n^2}{2}\right)$,

and $s_Q[n] \sim \mathcal{N}\left(s_{t,Q}[n], \frac{\sigma_n^2}{2}\right)$ (from (2.6) and (A.2)).

Let $X_k \sim \mathcal{N}\left(\mu_k, \frac{\sigma_n^2}{2}\right)$, for $k = 0, 1, \ldots, 2N-1$, be a sequence of random variables corresponding to $s_I[0], s_Q[0], s_I[1], s_Q[1], \ldots, s_I[N-1], s_Q[N-1]$. It follows that the mean of each random variable is based on the corresponding $s_t[n]$ signal value:

$$\mu_k = \begin{cases} s_{t,I}\left[\frac{k}{2}\right] & \text{for } k \text{ even}, \\ s_{t,Q}\left[\lfloor\frac{k}{2}\rfloor\right] & \text{for } k \text{ odd}. \end{cases} \tag{A.3}$$

Following on from (A.1):

$$\begin{aligned} \lambda_{norm} &= \sum_{n=0}^{N-1} \left(\frac{s_I[n]}{\left(\frac{\sigma_n}{\sqrt{2}}\right)} \right)^2 + \sum_{n=0}^{N-1} \left(\frac{s_Q[n]}{\left(\frac{\sigma_n}{\sqrt{2}}\right)} \right)^2 \\ &\equiv \sum_{k=0}^{2N-1} \left(\frac{X_k}{\left(\frac{\sigma_n}{\sqrt{2}}\right)} \right)^2 . \end{aligned} \tag{A.4}$$

By definition, λ_{norm} is therefore distributed according to a non-central χ^2-distribution with $2N$ DOF. Also by definition:

$$\begin{aligned} \delta &= \sum_{k=0}^{2N-1} \left(\frac{\mu_k}{\left(\frac{\sigma_n}{\sqrt{2}}\right)} \right)^2 \\ &= \frac{2}{\sigma_n^2} \sum_{n=0}^{N-1} \left(s_{t,I}^2[n] + s_{t,Q}^2[n] \right) \\ &= \frac{2N}{\sigma_n^2} \left(\frac{1}{N} \sum_{n=0}^{N-1} |s_t[n]|^2 \right) \\ &= 2N \times \text{SNR (see (2.13))}. \end{aligned} \tag{A.5}$$

A.3 Using the DFT In Lieu of Fourier Series Coefficients when Estimating the Cyclic Autocorrelation Magnitude

Here it is shown that, when the magnitude of the CAF is to be *estimated* for a discrete-time signal, the DFT can be used in lieu of evaluating (2.20), and a relationship exists between the indices of the DFT coefficients (k) and the corresponding indices of the Fourier Series coefficients (m).

The magnitude of the CAF of a discrete-time signal $s[n]$ of length N samples, from (2.26), is:

$$|R_s^\alpha[\nu]| = \left| \frac{1}{N} \sum_{n=0}^{N-1} s[n+\nu] s^*[n] e^{-j2\pi\alpha n T_s} \right| \text{ for sufficiently large } N. \quad \text{(A.6)}$$

The DFT of the delay product, $s[n+\nu]s^*[n]$, can be called $S_D[k]$, whose magnitude is, by definition:

$$|S_D[k]| = \left| \sum_{n=0}^{N-1} s[n+\nu] s^*[n] e^{-\frac{j2\pi kn}{N}} \right|. \quad \text{(A.7)}$$

When the scaling factor $\frac{1}{N}$ is ignored, $|S_D[k]| \equiv |R_s^\alpha[v]|$, when the following also holds:

$$\begin{aligned} \frac{k}{N} &= \alpha T_s = \frac{\alpha}{f_s} = \frac{\alpha}{N\Delta f} \\ \therefore k &= \frac{\alpha}{\Delta f}, \end{aligned} \quad \text{(A.8)}$$

where Δf is the *frequency resolution* of the DFT.

In the case where the observation length is an integer-number of symbols,

M, of an OFDM signal with a cyclic prefix:

$$N = MN_{symb}. \tag{A.9}$$

Therefore:

$$\Delta f = \frac{f_s}{N} = \frac{f_s}{MN_{symb}} = \frac{f_s}{M\left(\frac{T_{symb}}{T_s}\right)} = \frac{1}{MT_{symb}}. \tag{A.10}$$

Substituting this into (A.8) results in:

$$k = \alpha MT_{symb}. \tag{A.11}$$

Furthermore, for the particular case where $\nu = N_{symb}$, it is seen from section 2.3.1 that the fundamental cycle frequency is:

$$f_\alpha = \frac{1}{T_{symb}}. \tag{A.12}$$

Therefore, $\alpha = mf_\alpha = \frac{m}{T_{symb}}$, where m is the Fourier Series coefficient index. Substituting this into (A.11) gives the necessary relationship between the DFT coefficient index, k, and the index of the corresponding Fourier Series coefficient that it approximates:

$$k = mM. \tag{A.13}$$

A.4 Computational Complexity of a Full Cyclostationarity Detector using the FFT Accumulation Method on the Entire $f - \alpha$ Plane

In this section, the reasoning behind (2.44) is explained.

An assumption made here is that the number of complex multiplies for a P-point FFT is of the order $P \log_2 P$.

Firstly, the value of P used in the analysis needs to be determined – this is constrained by the selection of $M = \dfrac{\Delta f}{\Delta \alpha} = 2$:

$$\begin{aligned} M &= \frac{N_{fam}}{N'} = \frac{PL}{N'} \\ \therefore P &= \frac{MN'}{L} = \frac{2N'}{L} \end{aligned} \tag{A.14}$$

(from (2.35), (2.36) and (2.37)).

In a single SCD estimate, N_{fam} samples are needed. For an observation of length N, several such SCD estimates may be made (and accumulated or averaged, in order to use all the available signal) where $N > N_{fam}$, in which case the number of such SCD estimates required is given by:

$$m = \frac{N}{N_{fam}} = \frac{N}{MN'} = \frac{N}{2N'} \tag{A.15}$$

(using (A.14)). Therefore, if one FAM estimate requires C_{fam} complex multiplies, then:

$$C_S = mC_{fam}. \tag{A.16}$$

Initially, complex demodulates are obtained corresponding to each of the N' frequency bins, by taking an N'-point FFT P times (one for each window). The number of complex multiplies required is:

$$C_1 = PN' \log_2 N' = \left(\frac{2N'^2}{L}\right) \log_2 N'. \tag{A.17}$$

At each frequency bin, there is a complex demodulate sequence (which can be considered to be a vector) of length P, so a *set* of N' vectors results. A complete estimate using FAM requires an element-wise multiplication between every possible *permutation* of two vectors selected from this set (not necessarily unique). There are N'^2 such permutations, so the number of complex multiplies is:

$$C_2 = PN'^2 = \frac{2N'^3}{L}, \tag{A.18}$$

which results in N'^2 product vectors. Then, a P-point FFT has to be carried out on each of these vectors, which involves:

$$C_3 = N'^2 P \log_2 P = \frac{2N'^3}{L} \log_2 \left(\frac{2N'}{L}\right) \tag{A.19}$$

complex multiplies.

At this point, one complete SCD estimate has been obtained, so from (A.16):

$$\begin{aligned} C_{fam} &= C_1 + C_2 + C_3 = \frac{2N'^2}{L}\left(\log_2 N' + N'\left(1 + \log_2\left(\frac{2N'}{L}\right)\right)\right) \\ \therefore C_S &= \frac{NN'}{L}\left(\log_2 N' + N'\left(1 + \log_2\left(\frac{2N'}{L}\right)\right)\right). \end{aligned} \tag{A.20}$$

Obtaining the test statistic then involves finding the magnitude-squared of each SCD point estimate, represented by an element from each of the N'^2 product vectors. Assuming that this can be done by multiplying each point

estimate by its conjugate, the number of complex multiplies is then:

$$C_T = PN'^2 = \frac{2N'^3}{L}. \tag{A.21}$$

A.5 Computational Complexity of a Reduced Cyclostationarity Detector using the FFT Accumulation Method

In arriving at (2.45), the reasoning follows closely that in section A.4 – the difference is based on the reduced portion of the $f-\alpha$ plane for which SCD estimates are obtained, and the resulting summation in (2.38).

Firstly, it is observed that the area of the normalized $f-\alpha$ plane, as shown in FIGURE 2.7, is equal to 1. Then, the region of interest for which estimates are performed, Ω, is bounded by two rectangles (see FIGURE 2.9): one with width $\frac{20}{128}$ and height $\frac{6}{128}$, and the other with width $\frac{12}{128}$ and height $\frac{6}{128}$. Hence, Ω covers an area of $\frac{3}{256}$ - this scale factor affects C_2, C_3 and C_T.

Hence, for this case:

$$\begin{aligned} C_{fam} &= C_1 + \frac{3}{256}(C_2 + C_3) \\ &= \frac{2N'^2}{L}\left(\log_2 N' + \frac{3}{256}N'\left(1 + \log_2\left(\frac{2N'}{L}\right)\right)\right) \\ \therefore C_S &= \frac{NN'}{L}\left(\log_2 N' + \frac{3}{256}N'\left(1 + \log_2\left(\frac{2N'}{L}\right)\right)\right), \end{aligned} \tag{A.22}$$

and:

$$C_T = \frac{3}{256}PN'^2 = \frac{3N'^3}{128L}. \tag{A.23}$$

A.6 Probability Density Functions per Frequency Bin for PSD Estimates of a Special-Case OFDM Signal with Additive White Gaussian Noise

This section verifies the assertion in (4.27).

To begin with, consider the case where the licensed network user of interest is not transmitting, so the cognitive radio only sees AWGN across the sensing bandwidth. In this case, the total noise power across the sensing bandwidth, σ_w^2, is distributed equally between the in-phase and quadrature components of the observed signal, $s_{w,I}[n]$ and $s_{w,Q}[n]$ respectively:

$$s_w[n] = s_{w,I}[n] + s_{w,Q}[n], \tag{A.24}$$

and both components have Gaussian distributions:

$$s_{w,I}[n], s_{w,Q}[n] \sim \mathcal{N}\left(0, \frac{\sigma_w^2}{2}\right). \tag{A.25}$$

(4.26) can be re-written as:

$$\begin{aligned} S[k] &= \frac{2}{K\sigma_w^2}\left(\left(\mathrm{Re}\left\{\sum_{n=0}^{K-1} s_w[n]\, e^{-\frac{2\pi kn}{K}}\right\}\right)^2 + \left(\mathrm{Im}\left\{\sum_{n=0}^{K-1} s_w[n]\, e^{-\frac{2\pi kn}{K}}\right\}\right)^2\right) \\ &= \left(\frac{\sqrt{2}}{\sqrt{K}\sigma_w}\sum_{n=0}^{K-1}\mathrm{Re}\left\{s_w[n]\, e^{-\frac{2\pi kn}{K}}\right\}\right)^2 + \left(\frac{\sqrt{2}}{\sqrt{K}\sigma_w}\sum_{n=0}^{K-1}\mathrm{Im}\left\{s_w[n]\, e^{-\frac{2\pi kn}{K}}\right\}\right)^2. \end{aligned} \tag{A.26}$$

Expanding further:

$$\begin{aligned}\mathrm{Re}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &= s_{w,I}[n]\cos\left(\frac{2\pi kn}{K}\right) + s_{w,Q}[n]\sin\left(\frac{2\pi kn}{K}\right)\\ \mathrm{Im}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &= s_{w,Q}[n]\cos\left(\frac{2\pi kn}{K}\right) - s_{w,I}[n]\sin\left(\frac{2\pi kn}{K}\right).\end{aligned} \quad \text{(A.27)}$$

It is now noted that, for Gaussian random variables $X \sim \mathcal{N}\left(\mu_X, \sigma_X^2\right)$ and $Y \sim \mathcal{N}\left(\mu_Y, \sigma_Y^2\right)$, $cX \sim \mathcal{N}\left(c\mu_X, c^2\sigma_X^2\right)$ and $(X+Y) \sim \mathcal{N}\left(\mu_X+\mu_Y, \sigma_X^2+\sigma_Y^2\right)$, where c is a constant. The terms $\cos\left(\frac{2\pi kn}{K}\right)$ and $\sin\left(\frac{2\pi kn}{K}\right)$ may be treated as constants so that:

$$\begin{aligned}\mathrm{Re}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(0, \frac{\sigma_w^2}{2}\left(\cos^2\left(\frac{2\pi kn}{K}\right) + \sin^2\left(\frac{2\pi kn}{K}\right)\right)\right)\\ &\sim \mathcal{N}\left(0, \frac{\sigma_w^2}{2}\right) \quad [\text{since } \sin^2 x + \cos^2 x = 1],\end{aligned} \quad \text{(A.28)}$$

and similarly:

$$\mathrm{Im}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} \sim \mathcal{N}\left(0, \frac{\sigma_w^2}{2}\right). \quad \text{(A.29)}$$

When applying this result to (A.26), the following result becomes apparent:

$$\begin{aligned}\sum_{n=0}^{K-1}\mathrm{Re}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(0, \frac{K\sigma_w^2}{2}\right)\\ \sum_{n=0}^{K-1}\mathrm{Im}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(0, \frac{K\sigma_w^2}{2}\right),\end{aligned} \quad \text{(A.30)}$$

and furthermore:

$$\begin{aligned}\frac{\sqrt{2}}{\sqrt{K}\sigma_w}\sum_{n=0}^{K-1}\mathrm{Re}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(0,1\right)\\ \frac{\sqrt{2}}{\sqrt{K}\sigma_w}\sum_{n=0}^{K-1}\mathrm{Im}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(0,1\right).\end{aligned} \quad \text{(A.31)}$$

Applying this back to (A.26), it is seen that, in the absence of a transmission from a licensed network user, $S[k]$ is a random variable that, by definition, follows a central χ^2-distribution with 2 DOF – thus defining $f_0\left(S\left[k\right]\right)$.

Now, in the event of the licensed network user transmitting, the PSD estimate at some contiguous subset of the K frequency bins will follow a different pdf $f_1\left(S\left[k\right]\right)$ (the estimate associated with the remaining frequency bins will be unchanged). The simulation scenario in section 4.3.2 has been contrived so that, for the frequency bins occupied by the licensed network user's transmission, both the in-phase and quadrature signal magnitudes are equal to some constant value, m, in the DFT. Other than that, the variance of the PSD estimate values in these frequency bins remains the same due to the AWGN.

Therefore, for the occupied frequency bins, the following applies to the real and imaginary parts of the DFT value for that bin:

$$\begin{aligned}\sum_{n=0}^{K-1}\mathrm{Re}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(m,\frac{K\sigma_w^2}{2}\right)\\ \sum_{n=0}^{K-1}\mathrm{Im}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\} &\sim \mathcal{N}\left(m,\frac{K\sigma_w^2}{2}\right).\end{aligned} \tag{A.32}$$

If we divide both Gaussian random variables by their standard deviations, and then take the sum of the squares of both results:

$$\left(\frac{\sum_{n=0}^{K-1}\mathrm{Re}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\}}{\frac{\sqrt{K}\sigma_w}{2}}\right)^2+\left(\frac{\sum_{n=0}^{K-1}\mathrm{Im}\left\{s_w\left[n\right]e^{-\frac{2\pi kn}{K}}\right\}}{\frac{\sqrt{K}\sigma_w}{2}}\right)^2=S[k]. \tag{A.33}$$

So, by definition, the PSD estimates in the occupied frequency bins follow a non-central χ^2-distribution with 2 DOF, and the non-centrality parameter

is, also by definition:

$$\begin{aligned}\delta &= \left(\frac{m}{\left(\frac{\sqrt{K}\sigma_w}{\sqrt{2}}\right)}\right)^2 + \left(\frac{m}{\left(\frac{\sqrt{K}\sigma_w}{\sqrt{2}}\right)}\right)^2 \\ &= 4\left(\frac{m}{\sqrt{K}\sigma_w}\right)^2. \end{aligned} \tag{A.34}$$

This distribution thus defines $f_1\left(S\left[k\right]\right)$.

List of Abbreviations

ADC	Analogue-to-digital converter
ACMA	Australian Communications and Media Authority (Australia)
AWGN	Additive white Gaussian noise
BER	Bit error rate
BPF	Bandpass filter
CAD	Computer-aided design
CAF	Cyclic autocorrelation function
CDMA	Code division multiple access
CPR	Channel pair region
CPU	Central processing unit
CRN	Cognitive radio network
DARPA	Defense Advanced Research Projects Agency (USA)
DC	Duty cycle
DECT	Digital Enhanced Cordless Telecommunications
DFT	Discrete Fourier Transform
DTFT	Discrete-time Fourier Transform

DOF	Degrees of freedom
DS-CDMA	Direct sequence CDMA
DTV	Digital television
FAM	FFT Accumulation Method
FCC	Federal Communications Commission (USA)
FDMA	Frequency division multiple access
FFT	Fast Fourier Transform
FH-CDMA	Frequency hopping CDMA
FIFO	First in, first out
GA	Genetic algorithm
GPS	Global Positioning System
GSM	Global System for Mobile Communications
ICI	Inter-carrier interference
IDFT	Inverse Discrete Fourier Transform
IEEE	Institute of Electrical and Electronics Engineers
iid	Independent and identically distributed
INR	Interference-to-noise ratio
ISI	Inter-symbol interference
ISM	Industrial, Scientific and Medical
LZ	Lempel-Ziv (class of algorithms published by Abraham Lempel and Jacob Ziv)
LZC	Lempel-Ziv complexity
M2M	Machine-to-machine
MAP	Maximum a posteriori probability
MDP	Markov decision process

MIPS	Million instructions per second
OFDM	Orthogonal frequency division multiplexing
PSD	Power spectral density
pdf	Probability density function
pmf	Probability mass function
QoS	Quality of service
REM	Radio environment map
RF	Radio frequency
RX	Receive, Receiver
SCD	Spectral correlation density
SDR	Software-defined radio
SIMD	Single instruction, multiple data
SINR	Signal-to-interference-plus-noise ratio
SNR	Signal-to-noise ratio
SoC	System on a chip
SRRC	Square-root raised cosine
TDMA	Time division multiple access
TX	Transmit, Transmitter
UWB	Ultra-wideband
WRAN	Wireless regional area network

List of Symbols

x^*	Complex conjugate of x
$\mathbf{x}$	Lower-case with boldface indicates a vector quantity
$\|X\|$	Magnitude of a scalar X, or cardinality (size) of a set X
$\|\mathbf{x}\|$	Length of a vector $\mathbf{x}$
$x \vee y$	Logical OR of x and y
α	Cyclic frequency variable; or, the learning rate in reinforcement learning
χ^2	Chi-squared (distribution)
δ	Non-centrality parameter
γ	Discount factor in reinforcement learning
λ	Test statistic calculated by a signal detector
μ	Mean
σ	Standard deviation
σ^2	Variance
$E\{x\}$	Expected value of x
f	Continuous frequency variable

f_s	Sampling rate for discrete-time signals, equal to $\frac{1}{T_s}$
f_{symb}	Symbol rate
$f(y\|X = x)$	Conditional probability density function of quantity y, given random variable X has value x
H_0	The hypothesis that a signal of interest is absent from an observation
H_1	The hypothesis that a signal of interest is present in an observation
$\mathrm{Im}\{x\}$	Imaginary part of x
$\max\{x\}$	The maximum value of x
$\min\{x\|Y\}$	The minimum value of x such that condition Y is met
n	Discrete-time variable
$\binom{n}{k}$	The number of *combinations* of size k, drawn from a set of size n
N	Number of samples in a discrete-time signal observation; or, length of a sequence with a finite alphabet
$\mathcal{N}(\mu, \sigma^2)$	A Gaussian (normal) probability distribution with mean μ, variance σ^2
$P(E)$	The probability of an event E occurring
$\mathrm{Re}\{x\}$	Real part of x
$s[n]$	Discrete-time signal s
$s(t)$	Continuous-time signal s
T_s	Sampling period for discrete-time signals, equal to $\frac{1}{f_s}$

t	Continuous-time variable
t_n	The nth time epoch where a significant event occurs (e.g. a decision/action is taken)

References

[1] *ACMA communications report 2007-08*. Tech. rep., Australian Communications and Media Authority (2008).

[2] *ACMA communications report 2008-09*. Tech. rep., Australian Communications and Media Authority (2009).

[3] *ACMA communications report 2009-10*. Tech. rep., Australian Communications and Media Authority (2010).

[4] *ACMA communications report 2010-11*. Tech. rep., Australian Communications and Media Authority (2011).

[5] Australian Bureau of Statistics. *8153.0 - internet activity, Australia, Jun 2012 - type of access connection*. http://www.abs.gov.au/ausstats/abs@.nsf/Lookup/8153.0Chapter4Jun%202012 (2012).

[6] Australian Bureau of Statistics. *8153.0 - internet activity, Australia, Jun 2012 - volume of data downloaded*. http://www.abs.gov.au/ausstats/abs@.nsf/Lookup/8153.0Chapter8Jun%202012 (2012).

[7] Australian Bureau of Statistics. *8153.0 - internet activity, Australia, Jun 2012 - mobile handset subscribers*. http://www.abs.gov.au/ausstats/abs@.nsf/Lookup/8153.0Chapter9Jun%202012 (2012).

[8] P. Rysavy. *4G world: The need for more spectrum.* http://www.informationweek.com/mobility/3g/4g-world-the-need-for-more-spectrum/240012599 (2012).

[9] P. E. Ross. *When will software have the right stuff?* Spectrum, IEEE, **48**(12), 38 (2011).

[10] Australian Communications and Media Authority. *Radiocommunications licensing.* http://www.acma.gov.au/WEB/STANDARD/pc=PC_481 (2012).

[11] J. Mitola III. *Software radios: Survey, critical evaluation and future directions.* Aerospace and Electronic Systems Magazine, IEEE, **8**(4), 25 (1993).

[12] A. S. Margulies and J. Mitola III. *Software defined radios: a technical challenge and a migration strategy.* In *IEEE 5th International Symposium on Spread Spectrum Techniques and Applications, 1998*, vol. 2, pp. 551–556.

[13] P. Koch and R. Prasad. *The universal handset.* Spectrum, IEEE, **46**(4), 36 (2009).

[14] J. Mitola III and G. Q. Maguire Jr. *Cognitive radio: making software radios more personal.* Personal Communications, IEEE, **6**(4), 13 (1999).

[15] J. Mitola III. *Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio.* Doctor of Technology (2000).

[16] S. Haykin. *Cognitive radio: brain-empowered wireless communications.* Selected Areas in Communications, IEEE Journal on, **23**(2), 201 (2005).

[17] M. Wellens, J. Wu and P. Mähönen. *Evaluation of spectrum occupancy in indoor and outdoor scenario in the context of cognitive radio.* In *2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom), 2007*, pp. 420–427.

[18] R. I. C. Chiang, G. B. Rowe and K. W. Sowerby. *A quantitative analysis of spectral occupancy measurements for cognitive radio.* In *IEEE 65th Vehicular Technology Conference (VTC2007-Spring), 2007*, pp. 3016–3020.

[19] B. Hamdaoui, P. Venkatraman and M. Guizani. *Opportunistic exploitation of bandwidth resources through reinforcement learning.* In *IEEE Global Telecommunications Conference (GLOBECOM), 2009*, pp. 1–6.

[20] D. A. Roberson. *Structural support for cognitive radio system deployment.* In *2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom), 2007*, pp. 401–407.

[21] S. M. Mishra, S. ten Brink, R. Mahadevappa and R. W. Brodersen. *Cognitive technology for ultra-wideband/WiMax coexistence.* In *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2007*, pp. 179–186.

[22] A. M. Rateb, S. K. Yusof and N. Fisal. *Code rate adaptation for mitigating the effects of detect and avoid (DAA) mechanism on the performance of UWB systems.* In *IEEE International RF and Microwave Conference (RFM), 2008*, pp. 96–100.

[23] T. Yucek and H. Arslan. *A survey of spectrum sensing algorithms for cognitive radio applications.* Communications Surveys & Tutorials, IEEE, **11**(1), 116 (2009).

[24] S. Reisenfeld. *Performance bounds for detect and avoid signal sensing.* In *Second International Workshop on Cognitive Radio and Advanced Spectrum Management (CogART), 2009*, pp. 132–137 (2009).

[25] M. McHenry, E. Livsics, T. Nguyen and N. Majumdar. *XG dynamic spectrum access field test results [topics in radio communications].* Communications Magazine, IEEE, **45**(6), 51 (2007).

[26] Y. Zhao, S. Mao, J. O. Neel and J. H. Reed. *Performance evaluation of cognitive radios: Metrics, utility functions, and methodology.* Proceedings of the IEEE, **97**(4), 642 (2009).

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

[28] A. Sahai, R. Tandra, S. M. Mishra and N. Hoven. *Fundamental design tradeoffs in cognitive radio systems.* In *Proceedings of the first international workshop on Technology and policy for accessing spectrum 2006*, p. 2 (ACM, 1234390).

[29] N. Hao, J. K. Choi and S. J. Yoo. *Cooperative channel operation MAC protocol for WLAN based indoor cognitive radio network systems.* In *8th IEEE International Conference on Computer and Information Technology (CIT), 2008*, pp. 640–645.

[30] D. Singhal, M. K. Sharma and R. M. Garimella. *Energy efficient localization of primary users for avoiding interference in cognitive networks.* In *International Conference on Computer Communication and Informatics (ICCCI), 2012*, pp. 1–5.

[31] A. Prompijit, C. Sertthin, M. Nakagawa and T. Fujii. *Frequency sharing secondary system with carrier sense assisted by cellular system.*

In *3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom), 2008*, pp. 1–6.

[32] T. Maseng and T. Ulversoy. *Dynamic frequency broker and cognitive radio.* In *IET Seminar on Cognitive Radio and Software Defined Radios: Technologies and Techniques, 2008*, pp. 1–5.

[33] E. O. Nuallain. *A proposed propagation-based methodology with which to address the hidden node problem and security/reliability issues in cognitive radio.* In *4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 2008*, pp. 1–5.

[34] M. Barbiroli, C. Carciofi, A. Guidotti and D. Guiducci. *Evaluation and analysis of the hidden node margin for cognitive radio system operation in a real scenario.* In *Proceedings of the 5th European Conference on Antennas and Propagation (EUCAP), 2011*, pp. 1309–1313.

[35] O. Younis, D. Shallcross, L. Kant, K. Young, C. Graff and M. Patel. *TDMA scheduling and channel assignment for cognitive tactical networks.* In *Military Communications Conference (MILCOM), 2012*, pp. 1–6.

[36] A. Rahim Biswas, F. Berens, S. Zeisberg and A. Finger. *Determination of time domain mitigation parameters for coexistence of WiMedia and WiMax systems.* In *IEEE International Conference on Ultra-Wideband (ICUWB), 2008*, vol. 3, pp. 59–62.

[37] Z. Teng and C. Zhao. *Study on nonlinear variable period spectrum detection of cognitive radio.* In *5th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 2009*, pp. 1–3.

[38] H. Du, Z. Wei, Y. Yang and D. Yang. *Sensing overhead and average detection time mitigation for sensing scheduling algorithm.* In *18th International Conference on Telecommunications (ICT), 2011*, pp. 216–220.

[39] K. Chang, B. Senadji and V. Chandran. *Analysis of detection performance in spectrum sensing optimisation for long sensing periods.* In *IEEE 22nd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), 2011*, pp. 457–461.

[40] M. Di Felice, R. Doost-Mohammady, K. R. Chowdhury and L. Bononi. *Smart radios for smart vehicles: Cognitive vehicular networks.* Vehicular Technology Magazine, IEEE, **7**(2), 26 (2012).

[41] H. Li, G. Zhu, L. Jian, Z. Liang and D. Wang. *Stochastic spectrum access based on learning automata in cognitive radio network.* In *IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS), 2009*, vol. 3, pp. 294–298.

[42] I. Macaluso, T. K. Forde, L. DaSilva and L. Doyle. *Impact of cognitive radio: Recognition and informed exploitation of grey spectrum opportunities.* Vehicular Technology Magazine, IEEE, **7**(2), 85 (2012).

[43] A. Lempel and J. Ziv. *On the complexity of finite sequences.* Information Theory, IEEE Transactions on, **22**(1), 75 (1976).

[44] P. D. Sutton, K. E. Nolan and L. E. Doyle. *Cyclostationary signatures in practical cognitive radio applications.* Selected Areas in Communications, IEEE Journal on, **26**(1), 13 (2008).

[45] W. Wang. *Denial of service attacks in cognitive radio networks.* In *International Conference on Environmental Science and Information Application Technology (ESIAT), 2010*, vol. 2, pp. 530–533.

[46] Z. Luo, C. Lou, S. Chen, S. Zheng and S. Li. *Specific primary user sensing for wireless security in IEEE 802.22 network*. In *11th International Symposium on Communications and Information Technologies (ISCIT), 2011*, pp. 18–22.

[47] Y. Zhao, J. Gaeddert, L. Morales, K. Bae, J.-S. Um and J. H. Reed. *Development of radio environment map enabled case- and knowledge-based learning algorithms for IEEE 802.22 WRAN cognitive engines*. In *2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom), 2007*, pp. 44–49.

[48] H. Bezabih, B. Ellingsaeter, J. Noll and T. Maseng. *Digital broadcasting: Increasing the available white space spectrum using TV receiver information*. Vehicular Technology Magazine, IEEE, **7**(1), 24 (2012).

[49] *FCC 08-260: Second report and order and memorandum opinion and order*. November 14, 2008, Federal Communications Commission (2008).

[50] *Spectrum reallocation in the 700 MHz digital dividend band*. Tech. rep., Australian Communications and Media Authority (2010).

[51] *ACMA moves ahead with auction of spectrum in the 700 MHz (digital dividend) and 2.5 GHz bands*. http://www.acma.gov.au/WEB/STANDARD/pc=PC_312543 (2011).

[52] *Applications open for the April digital dividend auction*. http://www.acma.gov.au/WEB/STANDARD/pc=PC_600142 (2013).

[53] T. Bräysy, J. Lehtomaki, B. Calvet, S. Delmas and C. Moy. *Cognitive techniques for finding spectrum for public safety services*. In *Proceedings of Military Communications and Information Systems Conference, MCC'10* (2010).

[54] R. Ferrus, O. Sallent, G. Baldini and L. Goratti. *Public safety communications: Enhancement through cognitive radio and spectrum sharing principles*. Vehicular Technology Magazine, IEEE, **7**(2), 54 (2012).

[55] H. A. Karim, H. Mohamad, N. Ramli and A. Sali. *Scalable video streaming over overlay/underlay cognitive radio network*. In *International Symposium on Communications and Information Technologies (ISCIT), 2012*, pp. 668–672.

[56] S. Kandeepan, G. Baldini and R. Piesiewicz. *Experimentally detecting IEEE 802.11n Wi-Fi based on cyclostationarity features for ultra-wide band cognitive radios*. In *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, 2009*, pp. 2315–2319.

[57] L. Luo, N. M. Neihart, S. Roy and D. J. Allstot. *A two-stage sensing technique for dynamic spectrum access*. Wireless Communications, IEEE Transactions on, **8**(6), 3028 (2009).

[58] W. Zhang and Y. Sanada. *Dual-stage detection scheme for detect and avoid*. In *IEEE International Conference on Ultra-Wideband (ICUWB), 2010*, vol. 2, pp. 1–4.

[59] S. Yarkan and H. Arslan. *Binary time series approach to spectrum prediction for cognitive radio*. In *IEEE 66th Vehicular Technology Conference (VTC-2007 Fall), 2007*, pp. 1563–1567.

[60] U. Berthold, F. Fu, M. van der Schaar and F. K. Jondral. *Detection of spectral resources in cognitive radios using reinforcement learning*. In *3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2008*, pp. 1–5.

[61] H. Urkowitz. *Energy detection of unknown deterministic signals*. Proceedings of the IEEE, **55**(4), 523 (1967).

[62] H. Tang. *Some physical layer issues of wide-band cognitive radio systems.* In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2005*, pp. 151–159.

[63] W. A. Gardner. *Exploitation of spectral redundancy in cyclostationary signals.* Signal Processing Magazine, IEEE, **8**(2), 14 (1991).

[64] A. F. Lima Jr. *Analysis of low probability of intercept (LPI) radar signals using cyclostationary processing.* Masters thesis (2002).

[65] J. Wang, T. Chen and B. Huang. *Cyclo-period estimation for discrete-time cyclo-stationary signals.* Signal Processing, IEEE Transactions on, **54**(1), 83 (2006).

[66] H.-S. Chen, W. Gao and D. G. Daut. *Spectrum sensing using cyclostationary properties and application to IEEE 802.22 WRAN.* In *IEEE Global Telecommunications Conference (GLOBECOM '07), 2007*, pp. 3133–3138.

[67] W. A. Gardner. *Statistical Spectral Analysis: A Nonprobabilistic Theory* (Prentice Hall, Englewood Cliffs, New Jersey, 1987).

[68] R. van Nee and R. Prasad. *OFDM for Wireless Multimedia Communications.* Artech House universal personal communications library (Artech House, 2000).

[69] D. Panaitopol, R. Datta and G. Fettweis. *Cyclostationary detection of cognitive radio systems using GFDM modulation.* In *IEEE Wireless Communications and Networking Conference (WCNC), 2012*, pp. 930–934.

[70] R. S. Roberts, W. A. Brown and H. H. Loomis Jr. *Computationally efficient algorithms for cyclic spectral analysis.* Signal Processing Magazine, IEEE, **8**(2), 38 (1991).

[71] L. L. Hanzo, W. Webb and T. Keller. *Single and Multi-carrier Quadrature Amplitude Modulation: Principles and Applications for Personal Communications, WLANs and Broadcasting* (John Wiley & Sons, 2000).

[72] S. Chennakeshu and G. Saulnier. *Differential detection of $\pi/4$-shifted-DQPSK for digital cellular radio.* Vehicular Technology, IEEE Transactions on, **42**(1), 46 (1993).

[73] L. Gwennap. *Two-headed snapdragon takes flight: Qualcomm samples dual-CPU mobile processor at 1.2GHz.* Tech. rep. (2010).

[74] *Qualcomm products and services - the Snapdragon platform.* http://www.qualcomm.com/products_services/chipsets/snapdragon.html (2011).

[75] B. Klug and A. L. Shimpi. *Qualcomm's new Snapdragon S4: MSM8960 & Krait architecture explored.* http://www.anandtech.com/show/4940/qualcomm-new-snapdragon-s4-msm8960-krait-architecture (2011).

[76] B. Klug and A. L. Shimpi. *Qualcomm Snapdragon S4 (Krait) performance preview - 1.5 GHz MSM8960 MDP and Adreno 225 benchmarks.* http://www.anandtech.com/show/5559/qualcomm-snapdragon-s4-krait-performance-preview-msm8960-adreno-225-benchmarks (2012).

[77] *Processors & chipsets — mobile technology — Qualcomm Snapdragon processors.* http://www.qualcomm.com/snapdragon (2013).

[78] P. H. Winston. *Artificial Intelligence.* Addison-Wesley series in computer science (Addison-Wesley, Reading, Mass., 1992), 3rd ed.

[79] E. Alpaydın. *Introduction to Machine Learning* (MIT Press, 2004).

[80] K. Shanmugam and A. M. Breipohl. *An error correcting procedure for learning with an imperfect teacher.* Systems, Man and Cybernetics, IEEE Transactions on, **1**(3), 223 (1971).

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

[82] Y. Zeng, Y.-C. Liang and M. W. Chia. *Edge based wideband sensing for cognitive radio: Algorithm and performance evaluation.* In *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2011*, pp. 538–544 (2011).

[83] S. El-Khamy, M. El-Mahallawy and E. Youssef. *Improved wideband spectrum sensing techniques using wavelet-based edge detection for cognitive radio.* In *2013 International Conference on Computing, Networking and Communications (ICNC)*, pp. 418–423 (2013).

[84] T. M. Cover and J. A. Thomas. *Elements of Information Theory* (Wiley-Interscience, 1991).

[85] J. Ziv. *Coding theorems for individual sequences.* Information Theory, IEEE Transactions on, **24**(4), 405 (1978).

[86] M. Borowska, E. Oczeretko, A. Mazurek, A. Kitlas and P. Kuć. *Application of the Lempel-Ziv complexity measure to the analysis of biosignals and medical images.* Roczniki Akademii Medycznej w Bialymstoku – Annales Academiae Medicae Bialostocensis (Advances in Medical Sciences) Supplement 2, **50** (2005).

[87] *RWTH Aachen University static spectrum occupancy measurement campaign.* http://download.mobnets.rwth-aachen.de/ (2009).

[88] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning* (MIT Press, 1998).

[89] Q. Thai. *MATLAB Central - File Exchange - calc_lz_complexity.* http://www.mathworks.com.au/matlabcentral/fileexchange/38211-calclzcomplexity (2012).

[90] M. Wellens, J. Riihijärvi and P. Mähönen. *Empirical time and frequency domain models of spectrum use.* Physical Communication, **2**(1-2), 10 (2009). Elsevier Science Publishers B. V. Amsterdam, The Netherlands.